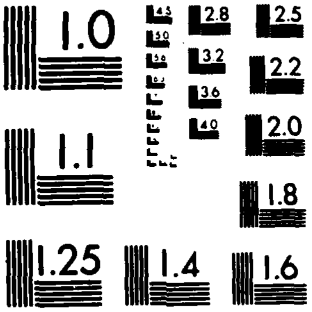
AD-A137 977 FEED (FIELD EXPLOITATION OF ELEVATION DATA) SOFTWARE DOCUMENTATION(U) GEORGIA INST OF TECH ATLANTA ENGINEERING EXPERIMENT STATION J G JAY ET AL. AUG 83

1/3

UNCLASSIFIED ETL-0335 DAAK70-82-K-0204 F/G 9/2 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

6

ETL–0335

FEED software documentation

J. G. Jay
E. A. Anderson

Georgia Institute of Technology
Engineering Experiment Station
Atlanta, Georgia 30332

August 1983

Prepared for

U.S. ARMY CORPS OF ENGINEERS
ENGINEER TOPOGRAPHIC LABORATORIES
FORT BELVOIR, VIRGINIA 22060

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

ETL

Destroy this report when no longer needed.
Do not return it to the originator.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The citation in this report of trade names of commercially available products does not constitute official endorsement or approval of the use of such products.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ETL-0335	2. GOVT ACCESSION NO. AD A137 977	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) FEED SOFTWARE DOCUMENTATION		5. TYPE OF REPORT & PERIOD COVERED Contract Report 8/24/82 — 1/31/83
		6. PERFORMING ORG. REPORT NUMBER A-3339
7. AUTHOR(s) J. G. Jay E. A. Anderson		8. CONTRACT OR GRANT NUMBER(s) DAAK70-82-K-0204
9. PERFORMING ORGANIZATION NAME AND ADDRESS Georgia Institute of Technology Engineering Experiment Station Atlanta, Georgia 30332		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army Engineer Topographic Laboratories Fort Belvoir, Virginia 22060		12. REPORT DATE August 1983
		13. NUMBER OF PAGES 262
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Digital, Elevation, FEED, Software, Terrain

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report provides programmer-level documentation which describes the FEED Software, structure, files, data, and procedures. → cont pg i

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) cont

PREFACE

The Georgia Institute of Technology Engineering Experiment Station ~~(GTEES)~~ is working with the U.S. Army Engineer Topographic Laboratories (ETL) to combine the existing capabilities of the FEED system with the elevation data algorithms and enhancements from the Digital Terrain Analysis Station ~~(DTAS)~~ to form a more generic, more machine-independent software system. Based on 16-bit minicomputer technology, this system will provide the Field Army with a capability for exploiting digital terrain elevation data and associated products by the first quarter of FY85. A follow-on capability in digital terrain elevation data products will be incorporated as part of the Terrain Analyst Work Station ~~(TAWS)~~, which is also being developed by ETL and will take advantage of 32-bit minicomputer technology and the concept of device-independent graphics.

This report documents the existing FEED software, which is the starting point for the development work referred to above. The FEED software is currently available for the Data General family of minicomputers using Tektronix PLOT10 graphics commands.

This report was generated under Contract DAAK70-82-K-0204 for the U.S. Army Engineer Topographic Laboratories, Fort Belvoir, Virginia 22060, by the Georgia Institute of Technology Engineering Experiment Station, Atlanta, Georgia 30332, and submitted as A-3339. The Contracting Officer's Representative was Mr. William Veigel.

A-1

CONTENTS

I. INTRODUCTION

The U.S. Army Engineer Topographic Laboratories (ETL) have developed the Field Exploitation of Elevation Data (FEED) system to generate terrain analysis graphics based on line-of-sight profile data. The system has been developed over a period of several years by a combination of in-house effort and contractor support. FEED has thus outlived the tenure of many software professionals. Although a user's manual exists along with explanation of the underlying engineering theory, a sufficient level of programmer documentation for the system has never been available.

FEED is a complex software system consisting of over 145 programs, subroutines, and functions, with the extensive use of overlays and program swaps. The ability of programmers to maintain, trouble-shoot, or enhance such software is absolutely dependent on the level of program documentation and software organization.

In the course of this project, programmer documentation for the FEED software has been produced. The purpose of each program has been identified, the parameters and COMMON variables have been described, the calling sequences and logic flow have been charted, and the structure and use of the disk data files have been recorded. In addition, the file organization on the FEED system disk pack has been improved so that the software source code found on the disk exactly matches current listings and documentation. Procedures for backing up the FEED disk to tape have been outlined.

At least one copy of this report is to be maintained in the form of a loose-leaf 3-ring binder. That document should continue to be updated and corrected each time any programmer corrects, updates, enhances, or in any other way changes any of the FEED software. Only in this manner will it be assured that the documentation always stays current and useful for maintaining this complex software system.

IA. CONTENTS OF THIS REPORT

Users of this report will reference three general subjects: program/subroutine descriptions, disk file structure and usage, and explanations of the common data areas. The contents of these sections of the manual include the following:

-Section II.A: A list of the executable programs and their functions, with a chart showing the program swapping sequence.

-Section II.B: An alphabetical list of all the subroutines in the FEED system, a brief description of each routine's function, the executable program in which this routine can be found, the corresponding diagram number, and a cross-reference as to which other routines use this subroutine as well as which other routines this subroutine uses.

-Section II.C: An expanded description of each routine, giving more detailed information on how the routine is used and explaining each parameter passed to and from the routine.

-Section II.D: Flow diagrams which chart the structure of the software system.

-Section III.A: An explanation of the disk file naming conventions, and a listing of which routines open, close, read, or write each file.

-Section III.B: A detailed description of the structure of each disk file.

-Section IV.A: A table showing which routines contain each common data area.

-Section IV.B: A description of how each common data area is used, with an explanation of the contents of each variable in common.

-Section IV.C: A table indicating the initial values of certain common variables which are set by different routines.

-Section V. An explanation of how the source files containig these routines are organized on the FEED disk packs.

-Section VI. Recommendations for disk and tape backup of the software.

IB. GLOSSARY

The following programming terms are used throughout this report. While it is expected that the primary users of this report will be computer programmers, to whom these terms are no doubt familiar ones, other readers may find a brief explanation useful.

EXECUTABLE PROGRAMS - These files, which are given the .SV extension by the Relocatable Loader (RLDR), are the programs in their final form ready for execution. The RLDR takes the compiled relocatable files (.RB) and builds the executable progarm. If the program is built using overlay segments, a corresponding .OL file is created.

FILE - Data, text, source code, etc. stored on the disk and assigned a unique name. File name extensions are used to identify specific types of files (i.e. .SV, .DB, .PF, etc).

LOAD MACRO - An ASCII file created with the .MC extension; contains the instructions to RLDR on which routines to include in each executable program.

LOGICAL UNIT NUMBER - An identifier assigned by the programmer inside the Fortran programs for each disk file. Different executable programs may attach different logical unit numbers to the same file.

MAIN PROGRAM - Each executable consists of a main program and one or more subroutines and libraries. The main program initiates execution, performs processing as required, and invokes the appropriate subroutines.

OVERLAY - When the size of an executable program exceeds the program memory space, the technique of overlaying may be used to reduce program size. The RLDR is instructed to load the program in such a manner that different program modules will share the same memory space; i.e. they execute at different times.

PARAMETERS - Variables passed to and from Fortran subroutines. There has to be a one-to-one correspondence between the calling routine and the subroutine as to the number of parameters and their data type.

RELOCATABLE FILES - These files (.RB) are created when the Fortran compiler complies the source code. The various .RB files are input to RLDR to build the executable programs.

RLDR - Relocatable Loader utility program provided by ROLM; used to link (i.e. build) the relocatable files into executable programs.

SOURCE FILES - These are text files entered directly by the programmer, which contain the Fortran commands and statements. These files are maintained using the text editor and are the files submitted to the compiler.

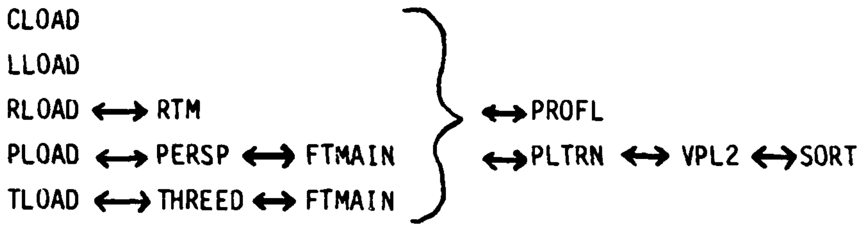
SUBROUTINE - Program modules which perform a specific task for one or more calling routines. Rather than duplicating source code in several places throughout a program, the code can be placed in one surboutine and then loaded into the appropriate executable programs. A FUNCTION is a special purpose subroutine, which is invoked without the use the Fortran CALL statement.

SWAP - Swapping is a technique for having one executable program invoke another executable program. When the second program is finished executing, it can swap back to the original program. In such a way, program execution, which otherwise would be too large to fit in available memory, can be accomplished by sequentially sharing the program space.

II. PROGRAM AND SUBROUTINE DESCRIPTIONS

IIA. LIST OF EXECUTABLE PROGRAMS

PRIMARY PROGRAMS


CLOAD

LLOAD

RLOAD ↔ RTM

PLOAD ↔ PERSP ↔ FTMAIN

TLOAD ↔ THREED ↔ FTMAIN

↔PROFL

↔PLTRN ↔ VPL2 ↔SORT

↔ = Program Swap

STAND ALONE PROGRAMS

MAINFT

DOPARM

FUNCTION

CLOAD - Produces contour plots.

DOPARM - Prints out plot parameters stored in plot file.

LLOAD - Produce line of sight plots.

FTMAIN - Feature plotting program swapped to for perspective and three dimensional plots.

PROFL - Selects the terrain profiles for each of the five primary FEED programs.

MAINFT - Creates, modifies, deletes feature data files.

PLOAD - Produces perspective plots.

RLOAD - Produces radar terrain masked plots.

SORT - Versatec supplied program - plots the output produced by VPL2.

TLOAD - Produces three dimensional plots.

VPL2 - Makes all the plotting calls for Versatec plotting. Swapped to by PLTRN.

PERSP - Swapped to by PLOAD - does the actual perspective plotting.

PLTRN - Produces the plots for each of the five primary FEED programs.

RTM - Swapped to by RLOAD - does the actual RTM plotting.

THREED - Swapped to by TLOAD - does actual 3-D plotting.

IIB. BRIEF PROGRAM DESCRIPTIONS

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
ADDREC	add feature data records to specified file; change location info. of records in file. Calls: DMSSEC,ERRFT,MGCORD, MOVEAA,ONEPNT,PRINR Called by: CHANGE, MAINFT	MAINFT	13
AGAIN	asks user if he wants another plot of the same type. Calls: nothing Called by: CMAIN,LMAIN, PMAIN,RMAIN,TMAIN	CLOAD LLOAD PLOAD RLOAD TLOAD	1 2 3 4 5
ALT	finds elevation of given location specified by lat. and long. Uses polynomial database. Calls: IUNPCK,MOVEAA,MOVEKA Called by: PTS,PTSSEC	PROFL	6
NALT	uses gridded database. Calls: nothing Called by: PTS,PTSSEC	PROFL	6
CALCPR	calc. profile and point numbers for a location in a perspective plot. Calls: nothing Called by: PRSBD,PRSFT	FTMAIN	11

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
CALCTH	calculates profile and point numbers for a location in a 3-D plot. Calls: nothing Called by: THRBD,THRFT	FTMAIN	11
CHANGE	change record code, description or location info. in feature data record. Calls: ADDREC,ERRFT,PRINR Called by: MAINFT	MAINFT	13
CMAIN	Main program for contour maps. Calls: AGAIN,COPSM,CPLOT, DRIVFT,INCON,MOVEAA,NPAGE,PLOT, PLTSV,PMPRT,PRCEED,WPRFRT,PSWAP, SETUP,TITLE Called by: nothing	CLOAD	1
CNVTR	driver routine for conversion routines LL2UTM and UTM2LL. Calls: LL2UTM,UTM2LL Called by: PTMN	PROFL	6
CON	plots contour levels between two profiles. Calls: PLOT,SYMBOL Called by: CPLOT,RPLOT	CLOAD RTM	1 8

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
VCON	version of CON used by perspective and 3-D plots Calls: PLOT,SYMBOL Called by: PPLOT,RPLOT	THREED PERSP	9 7
CONPLT	initial plotting for contour plots; scale factors, plot parameters, title etc. Calls: CPARSM,NPAGE,NUMBER,PLOT, SYMBOL Called by: CPLOT	CLOAD	1
CONTBD	plots boundary type feature data on contour map output. Calls: MOVEAA,PLOT,SYMBOL Called by: PLBPFT	CLOAD	12
CONTFT	Plots single point feature record on contour plot. Calls: SPSYM Called by: PLSPFT	CLOAD	12
COPSM	Puts command in plot file for Tektronix hard copy. Calls: PLOTB Called by: CMAIN,LOSPLT, PERSP,RTM,THREED	CLOAD LLOAD PERSP RTM THREED	1 2 7 8 9
COPY	sends hard copy request to Tektronix. Calls: Tektronix routines Called by: PL2	PLTRN	16,10

		EXECUTABLE	
FILE	FUNCTION	PROGRAM(S)	DIAGRAM
CPARSM	stores scaled contour plot parameters in plot file. Calls: PLOTB Called by: CONPLT	CLOAD	1
CPLOT	does plotting for contour map prog. Calls: CON,CONPLT,MOVEAA,MOVEKA,PRFRD Called by: CMAIN	CLOAD	1
DASH	plots dashed line to a point xz,yz with NUMDPI dashes per inch. Calls: PLOT Called by: PL2	PLTRN	10
DMSSEC	converts lat. or long. in deg., min., sec., dir., to signed seconds. Calls: nothing Called by: ADDREC,INCON, INLOS,INPRS,INRTM,INTHRD	CLOAD LLOAD MAINFT PLOAD RLOAD TLOAD	1 2 13 3 4 5
DOPARM	lists out plot file parameters. Uses .PM file. Calls: PLOTPM Called by: nothing	DOPARM	none
DRFTPR	driver to plot feature data on perspective plots. Calls: FTOPEN,LL2UTM,PRSBD, PRSFT,UTM2LL Called by: FTMAIN	FTMAIN	11

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
DRFTTH	driver to plot feature data on 3-D plots. Calls: FTOPEN,LL2UTM,THRBD, THRFT,UTM2LL Called by: FTMAIN	FTMAIN	11
DRIVFT	driver routine for feature data plots for contour and radar terrain mask. Calls: PLBPFT,PLSPFT,RDHDFT, SCALFT Called by: CMAIN,RTM	CLOAD RTM	1 8 12
DSHSM	stores parameters needed to plot dashed lines for line of sight. Calls: PLOTB Called by: LOSPLT	LLOAD	2
ERRFT	reports error code for MAINFT. Calls: nothing Called by: ADDREC,CHANGE,MAINFT, PRINFT,SRCHFI	MAINFT	13
ERTOPT	reads, validates, stores in common block the input for elevation correction option. Calls: nothing Called by: INLOS,INPRS, INRTM,INTHRD	LLOAD PLOAD RLOAD TLOAD	2 3 4 5

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
ERTPLT	writes the earth correction option on a plot. Calls: SYMBOL Called by: LOSPLT,RTMPLT	LLOAD RTM	2 8
ERTPRT	prints earth correction option stored in common block. Calls: nothing Called by: INLOS,INPRS, INRTM,INTHRD	LLOAD PLOAD RLOAD TLOAD	2 3 4 5
FTMAIN	driver for the feature plotting prog. swapped to from perspective and 3-D. Calls: DRFTPR,DRFTTH,GRDPRS,GRDTHR, MOVEAA,SCALPR,SCALTH Swapped to by FTSWP Called by: nothing	FTMAIN	11 7 9
FTOPEN	opens feature data file, reads 1st block, header record. Calls: nothing Called by: DRFTPR,DRFTTH	FTMAIN	11
FTSWP	fills file "PNTFL" with common blocks, swaps to "FTMAIN.SV" for feature and grid line plotting. Calls: MOVEAA Called by: PERSP,THREED	PERSP THREED	7 9

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
GRDPRS	plots grid lines for perspective view. Uses point file from PPLOT. Calls: PRSLIN Called by: FTMAIN	FTMAIN	11
GRDRTM	plots grid lines specified by IGRID on the RTM plots. Calls: PLOT Called by: RPLOT	RTM	8
GRDTHR	plots grid lines on 3-D plot. Uses point file created in TPLOT. Calls: PLOT,PTRD Called by: FTMAIN	FTMAIN	11
HIDDEN	operations on the hidden point matrix. Calls: MOVEKA Called by: PPLOT,PRSBD,PRSFT, THRBD,THRFT,TPLOT	FTMAIN PERSP THREED	11 7 9
IAZCHK	tests if a periodic value is within specified region. Function. Calls: nothing Called by: CALCPR,GRDPRS,GRDRTM, PRSBD,PTMN,RTMBD,RTMFT	FTMAIN RTM PROFL CLOAD	11 8 6 12
ICHK	tests if a value is between two other values. Function. Calls: nothing Called by: CONTBD, PRSBD,PRSLIN RTMBD, THRBD	CLOAD FTMAIN	12 11

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
INCON	reads, validates, prints inputs for contour plot option. Calls: DMSSEC,LL2UTM,MGBOUN, MOVEAA,NPAGE,PLTPRT,UTM2LL Called by: CMAIN	CLOAD	1
INLOS	reads, validates inputs for line of sight profile. Calls: DMSSEC,ERTOPT,ERTPRT, LL2UTM,MGCORD,MOVEAA,NPAGE,UTM2LL Called by: LMAIN	LLOAD	2
INPRS	reads, validates, prints inputs for the perspective plot. Calls: DMSSEC,ERTOPT,ERTPRT, LL2UTM,MGCORD,MOVEAA,NPAGE,PLTOPT, PLTPRT,RDGPRT,UTM2LL Called by: PMAIN	PLOAD	3
INRTM	reads, validates, prints inputs for the radar terrain mask plots. Calls: DMSSEC,ERTOPT,ERTPRT,LL2UTM, MGCORD,MOVEAA,NPAGE,PLTPRT,UTM2LL Called by: RMAIN	RLOAD	4
INTHRD	reads, validates, prints inputs for the 3-D plot option. Calls: DMSSEC,ERTOPT,ERTPRT, LL2UTM,MGBOUN,MOVEAA,NPAGE,PLTOPT, PLTPRT,RDGPRT,UTM2LL Called by: TMAIN	TLOAD	5

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
INVM	computes azimuth (radians) and dist (m) between 2 points, given the points coords in radians. Calls: SODINV Called by: LMAIN	LLOAD	2
IUNPCK	unpacks computer word into 2 integers, i.e. left byte and right byte. Calls: nothing Called by: TITLE,ALT,KAM2AS	CLOAD LLOAD PLOAD RLOAD TLOAD PROFL PLTRN	1 2 3 4 5 6 15
KAM2AS	converts alphanumeric array (A2 format) to array of ASCII decimal equiv. integers (ADE). for output to Tektronix terminal. Calls: IUNPCK Called by: SYMBOL	PLTRN	15
LL2UTM	Lat., long. to UTM conversion (major and minor zone). Calls: nothing Called by: CNVTR,DRFTPR,DRFTTH, INCON,INLOS,INPRS, INRTM,INTHRD,PLBPFT, PLSPFT,TITLE	CLOAD FTMAIN LLOAD PLOAD PROFL RLOAD RTM TLOAD	1 11 2 3 6 4 12 5

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
LMAIN	main prog. for line of sight plot. Calls: AGAIN,INLOS,INVM, LOSPLT,LOSPRT,MOVEAA,NPAGE, PMPRT,PRCEED,PRFRD,JPRFRT, SETUP,TITLE,UTM2LL Called by: nothing	LLOAD	2
LOSPLT	plots line of sight profile in 8" x 12" area. Calls: COPSM,DASH,ERTPLT,LPARSM,MOVEAA, MOVEKA,NUMBER,PLOT,PLTSV,PSWAP, SYMBOL Called by: LMAIN	LLOAD	2
LOSPRT	prints table of elevation values. Calls: NPAGE Called by: LMAIN	LLOAD	2
LPARSM	stores line of sight plot parameters in plot file. Calls: PLOTB Called by: LOSPLT	LLOAD	2
MAINFT	create, modify, delete feature data files Calls: ADDREC,CHANGE,DMSSEC,ERRFT, MGSET,MOVEAA,MOVEKA,ONEPNT, PRINFT,SRCHFI Called by: nothing	MAINFT	13
MGBOUN	user inputs boundary values in mil grid form. Calls: nothing Called by: INCON,INTHRD	CLOAD TLOAD	1 5

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
MGCORD	user inputs UTM value in mil	LLOAD	2
	grid form (EEEENNNN).	MAINFT	13
	Calls: nothing	PLOAD	3
	Called by: ADDREC,INLOS,	RLOAD	4
	INPRS,INRTM		
MGSET	Sets up prefix value for use on	CLOAD	1
	UTM coords. Sets up min possible	LLOAD	2
	values for northing and easting.	MAINFT	13
	Calls: nothing	PLOAD	3
	Called by: MAINFT,SETUP	RLOAD	4
		TLOAD	5
MOVEAA	moves the 1st NUM values in	ALL	
	the array IFROM to the 1st		
	NUM elements of the array ITO.		
	Calls: nothing		
	Called by: many programs		
MOVEKA	fills the 1st NUM values of	ALL	
	array ITO with the value IVAL.		
	Calls: nothing		
	Called by: many programs		
NPAGE	notifies user to make hard	ALL	
	copies and erase screen.		
	Calls: nothing		
	Called by: many programs		
NUMBER	Version 1-records plot	CLOAD	1
	command on plot file	LLOAD	2
	using PLOTB.	PERSP	7
	Calls: PLOTB	THREED	9
	Called by: many programs	RTM	8

		EXECUTABLE	
FILE	FUNCTION	PROGRAM(S)	DIAGRAM
	Version 2 - actually sends plot command to Tektronix. Calls: Tektronix routines Called by: PLTRN	PLTRN	15,10
ONEPNT	in future, will be subroutine to digitize data from maps; presently returns zeros. (not yet implemented) Calls: THMBPT Called by: ADDREC,MAINFT	MAINFT	13
PERSP	does actual plotting for perspective view. Swapped to from "PMAIN.SV" Calls: COPSM,FTSWP,MOVEAA,PLOT, PPLOT,PSWAP Swapped to by PRSWP	PERSP	7 3
PLBPFT	calc. position of each feature boundary point, converts it (if necessary), and calls plot routine. Calls: CONTBD,LL2UTM,RTMBD,UTM2LL Called by: DRIVFT	CLOAD RTM	12
PLOT	Version 1 - records plot command in plot file using PLOTB. Calls: PLOTB Called by: many programs	CLOAD LLOAD RTM PERSP THREED FTMAIN	1 2 8 7 9 11 12

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
	Version 2 - actually sends plot command to Tektronix. Calls: Tektronix routines Called by: PLTRN	PLTRN	10,15
PLOTB	stores all plot commands in buffer and writes them into plot file. Calls: MOVEAA,MOVEKA Called by: COPSM,CPARSM,DSHSM, LPARSM,PPARSM,RPARSM,TPARSM,PLOT, NUMBER,SYMBOL	CLOAD FTMAIN LLOAD PERSP RTM THREED	1 11 2 7 8 9 12
PLOTPM	prints parameters from the .PM file. Calls: MOVEAA Called by: DOPARM,PLTRN	DOPARM PLTRN	 10
PLOTS	plot initialization routine, defines plot file buffer, output device. Calls: Tektronix routines Called by: PLTRN	PLTRN	10
PLSPFT	calc. location of feature data point, converts it (if necess.), calls plot routine. Calls: CONTFT,LL2UTM,RTMFT,UTM2LL Called by: DRIVFT	CLOAD RTM	12

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
PLTOPT	reads, validates plot options used for perspective and 3-D plots. Calls: nothing Called by: INPRS,INTHRD	PLOAD TLOAD	3 5
PLTPRT	prints plot options chosen by user. Calls: nothing Called by: INCON,INPRS, INRTM,INTHRD	CLOAD PLOAD RLOAD TLOAD	1 3 4 5
PLTRN	actually does all the plotting produced by the other programs, reads plotting requests from plot file and invokes Tektronix routines. Calls: MOVEAA,NPAGE,PLOTPM,PLOTS, PL2,VPRMS,V2SWP Swapped to by PSWAP	PLTRN	10 1 2 7 8 9
PLTSV	gives user the option of saving a plot output file. Calls: MOVEAA Called by: CMAIN,LOSPLT, PRSWP,RTSWP,THSWP	CLOAD LLOAD PLOAD RLOAD TLOAD	1 2 3 4 5
PL2	Overlay segment where all plotting calls are made. Opens, reads, closes plot file. Calls: COPY,DASH,MOVEAA,MOVEKA, NUMBER,PLOT,SYMBOL,TSEND	PLTRN	10

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
PMAIN	main prog. for perspective plots Calls: AGAIN,INPRS,MOVEAA, NPAGE,PMPRT,PRCEED,JPRFRT, PRSWP,SETUP,TITLE Called by: nothing	PLOAD	3
PMPRT	puts a parameter file name into the parameter print file. Calls: MOVEAA Called by: CMAIN,LMAIN, PMAIN,RMAIN,TMAIN	CLOAD LLOAD PLOAD RLOAD TLOAD	1 2 3 4 5
PPARSM	stores parameters needed to do a perspective plot to scale. Calls: PLOTB Called by: PRSPLT	PERSP	7
PPLOT	does plotting for the perspective plot program. Calls: CON,HIDDEN,MOVEAA,MOVEKA, NUMBER,PLOT,JPRFRD,PRSPLT Called by: PERSP	PERSP	7
PRCEED	asks user if he wants to proceed with the inputs he is using. Calls: nothing Called by: CMAIN,LMAIN, PMAIN,RMAIN,TMAIN	CLOAD LLOAD PLOAD RLOAD TLOAD	1 2 3 4 5
PRFRD	reads profile from file created by PROFL Calls: MOVEAA Called by: CPLOT,LMAIN, TPLOT	CLOAD LLOAD THREED	1 2 9

		EXECUTABLE	
FILE	FUNCTION	PROGRAM(S)	DIAGRAM
JPRFRD	version of profile read used by perspective and radar terrain mask Calls: MOVEAA Called by: PPLOT,RPLOT	PERSP RTM	7 8
JPRFRT	creates or retrieves profiles needed by calling subroutine; swaps to PROFL prog. if profile file being created. Calls: MOVEAA,NPAGE,STAT Called by: PMAIN,RMAIN, LMAIN,TMAIN	LLOAD PLOAD RLOAD TLOAD	2 3 4 5
WPRFRT	version of profile retrieve used by contour. Calls: MOVEAA,NPAGE,STAT Called by: CMAIN	CLOAD	1
PRFWRT	writes profiles onto file in a buffered way to reduce the number of reads and writes. Calls: MOVEAA,MOVEKA Called by: PTMN	PROFL	6
PRINFT	print feature data file in readable format. Calls: ERRFT,SECDMS,UTM2MG Called by: MAINFT	MAINFT	13

		EXECUTABLE	
FILE	FUNCTION	PROGRAM(S)	DIAGRAM
PRINR	print feature data record in clear format. Calls: SECDMS,UTM2MG Called by: ADDREC,CHANGE	MAINFT	13
PROFL	determine terrain profiles requested by calling prog. writes profiles into profile file which calling prog. reads. Calls: MOVEAA,PTMN Swapped to by: JPRFRT and WPRFRT	PROFL	6
PRSBD	plots boundary features on perspective views. Uses point file and hidden matrix from PPLOT. Calls: CALCPR,HIDDEN,PLOT, PRSLIN,PTCALC,SYMBOL Called by: DRFTPR	FTMAIN	11
PRSFT	plots single point feature on perspective plots; uses point file and hidden matrix from PPLOT. Calls: CALCPR,HIDDEN,PTCALC, SPSYM Called by: DRFTPR	FTMAIN	11

		EXECUTABLE	
FILE	FUNCTION	PROGRAM(S)	DIAGRAM
PRSLIN	plots feature line between 2 points on a perspective plot. Uses point file from PPLOT. Calls: PLOT,PTCALC Called by: GROPRS,PRSBD	FTMAIN	11
PRSPLT	does initial plotting for the perspective plot. Calls: NUMBER,PLOT,PPARSM, SYMBOL Called by: PPLOT	PERSP	7
PRSWP	puts info. into swap file; swaps to "PERSP.SV". Calls: MOVEAA,NPAGE,PLTSV Called by: PMAIN	PLOAD	3
PSWAP	writes final block and header block of plot file;swaps to plot prog., PLTRN.SV. Calls: MOVEAA,MOVEKA,NPAGE Called by: CMAIN,LOSPLT, PERSP,RTM,THREED	CLOAD FTMAIN LLOAD PERSP RTM THREED	1 11 2 7 8 9
PTCALC	calc. screen posit. given profile point posit. Uses point file from PPLOT or TPLOT. Calls: PTRD Called by: PRSBD,PRSFT,PRSLIN, THRBD,THRFT	FTMAIN	11

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
PTMN	extracts the profiles for PROFL, and writes profiles to profile file. Calls: CNVTR,MOVEAA,MOVEKA, PRFWRT,PTSDR,SODDIR,WINDOW Called by: PROFL	PROFL	6
PTRD	reads into core part of the point file created in TPLOT or PPLOT. Calls: MOVEAA Called by: GRDTHR,PTCALC	FTMAIN	11
PTS	fills the PTS array, B, with elevations along a profile. Calls: ALT,NALT,SODDIR Called by: PTSDR	PROFL	6
PTSDR	overlay driver for subroutines PTS and PTSSEC Calls: PTS,PTSSEC Called by: PTMN	PROFL	6
PTSSEC	fills array, E, with elevations along a profile with const. lat. or long. for contour map or 3-D plot. Calls: ALT,NALT,MOVEKA Called by: PTSDR	PROFL	6
RDGPRT	prints the ridge line option chosen by the user. Calls: nothing Called by: INPRS,INTHRD	PLOAD TLOAD	3 5

		EXECUTABLE	
FILE	FUNCTION	PROGRAM(S)	DIAGRAM
RDHDFT	reads feature file header record to obtain (1) the # of records in file, (2) data conversion flag. Calls: nothing Called by: DRIVFT	CLOAD RTM	12
RMAIN	main prog. for radar terrain mask plots. Calls: AGAIN,INRTM,MOVEAA, NPAGE,PMPRT,PRCEED,JPRFRT, RTSWP,SETUP,TITLE Called by: nothing	RLOAD	4
RPARSM	stores parameters needed to do radar terrain plot to scale. Calls: PLOTB Called by: RTMPLT	RTM	8
RPLOT	does plotting for radar terrain mask prog. Calls: CON,GRDRTM,MOVEAA,MOVEKA, NUMBER,PLOT,JPRFRD,RTMPLT, SYMBOL,THKPLT Called by: RTM	RTM	8
RTM	does actual plotting for radar terrain mask. Swapped to from RMAIN Calls: COPSM,DRIVFT,MOVEAA, MOVEKA,PLOT,PSWAP,RPLOT Swapped to by: RTSWP	RTM	8 4

		EXECUTABLE	
FILE	FUNCTION	PROGRAM(S)	DIAGRAM
RTMBD	plots boundary type feature data on radar terrain masking output. Calls: MOVEAA,PLOT,SYMBOL Called by: PLBPFT	RTM	12
RTMFT	plots feature data single point record value on radar terrain masking plot. Calls: SPSYM Called by: PLSPFT	RTM	12
RTMPLT	does initial plotting for radar terrain mask option. Title, contour levels, etc. Calls: ERTPLT,NUMBER,PLOT, RPARSM,SYMBOL Called by: RPLOT	RTM	8
RTSWP	puts info. in swap file; swaps to "RTM.SV". Calls: MOVEAA,NPAGE,PLTSV Called by: RMAIN	RLOAD	4
SCALFT	calc. plot scale for a contour plot; calc. boundary values for 3-D plot. OBSOLETE - should be removed in future versions. Calls: nothing Called by: DRIVFT	CLOAD RTM	12

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
SCALPR	calc. parameters needed to plot grid lines and feature data on perspective plots. Calls: nothing Called by: FTMAIN	FTMAIN	11
SCALTH	calc. parameters needed to plot grid lines on 3-D plots. Calls: nothing Called by: FTMAIN	FTMAIN	11
SECDMS	converts lat. or long. in signed seconds to degrees, minutes, seconds and direction. Calls: nothing Called by: PRINFT,PRINR,TITLE	CLOAD LLOAD MAINFT PLOAD RLOAD TLOAD	1 2 13 3 4 5
SETUP	User inputs (1) hard copy option, (2) name of data base file. Calls: MGSET Called by: CMAIN,LMAIN, PMAIN,RMAIN,TMAIN	CLOAD LLOAD PLOAD RLOAD TLOAD	1 2 3 4 5
SODDIR	given a ref. point in radians, an azimuth in radians, and a dist. from the ref. pt. along the azimuth in meters, computes position of the new pt. in radians. Calls: nothing Called by: PTMN,PTS	PROFL	6

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
SODINV	computes parameters used by INVM in calculating azimuth and geodetic distance. Calls: nothing Called by: INVM	LLOAD	2
SPSYM	plots military symbols for single point feature data. Calls: PLOT,SZPLOT Called by: CONTFT,PRSFT,RTMFT, THRFT	CLOAD FTMAIN RTM	12 11
SRCHFI	search feature data file for location of a record in the file. Calls: ERRFT Called by: MAINFT	MAINFT	13
STAT	prints statistics on use of data base for any output. Zeros counters for next output. Calls: nothing Called by: JPRFRT,WPRFRT	CLOAD LLOAD PLOAD RLOAD TLOAD	1 2 3 4 5
SYMBOL	Version 1 - records plot command in plot file using PLOTB. Calls: PLOTB Called by: many programs	CLOAD LLOAD PERSP THREED RTM FTMAIN	1 2 7 9 8 11 12

	Version 2 - actually sends plot command to Tektronix Calls: Tektronix routines Called by: PLTRN	PLTRN	10
SZPLOT	plots the unit size symbol above the unit symbol already plotted. Calls: SYMBOL Called by: SPSYM,UNPLT	CLOAD FTMAIN RTM	12 11
THKPLT	plots a line of thickness TH from (X1, Y1) to (X2, Y2). Calls: PLOT Called by: RPLOT	RTM	8
THMBPT	used for digitizing the location of feature data. Calls: Tektronix routines Called by: ONEPNT	MAINFT	13
THRBD	plots boundary type feature data on 3-D plots. Calls: CALCTH,HIDDEN,MOVEAA,PLOT, PTCALC,SYMBOL Called by: DRFTTH	FTMAIN	11
THREED	does actuall plotting for oblique view (i.e. 3-D). Calls: COPSM,FTSWP,MOVEAA,PLOT, PSWAP,TPLOT Swapped to by: THSWP	THREED	9 5

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
THRFT	plots single-point feature data on 3-D plots. Calls: CALCTH,HIDDEN,PTCALL,SPSYM Called by: DRFTTH	FTMAIN	11
THREED	does actual plotting for oblique view (i.e. 3-D). Calls: COPSM,FTSWP,MOVEAA,PLOT, PSWAP,TPLOT Swapped to by: THSWP	THREED	9
THRFT	plots single-point feature data on 3-D plots. Calls: CALCTH,HIDDEN,PTCALC,SPSYM Called by: DRFTTH	FTMAIN	11
THRPLT	calc. plot scale, and projection plane tilt angle and does initial plotting for 3-D. Calls: PLOT,SYMBOL,TPARSM Called by: TPLOT	THREED	9
THSWP	puts info. into swap file; swaps to "THREED.SV" Calls: MOVEAA,NPAGE,PLTSV Called by: TMAIN	TLOAD	5

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
TITLE	prints title page at beginning of run; prints polynomial parameters if disk file present, prints area parameters if these are in core. Calls: IUNPCK,LL2UTM,MOVEAA,NPAGE, SECDMS,UTMPRT Called by: CMAIN,LMAIN,PMAIN, RMAIN,TMAIN	CLOAD LLOAD PLOAD RLOAD TLOAD	1 2 3 4 5
TMAIN	main prog. for 3-D or oblique view. (projection of area onto plane). Calls: AGAIN,INTHRD,MOVEAA, NPAGE,PMPRT,PRCEED, JPRFRT,SETUP,THSWP,TITLE Called by: nothing	TLOAD	5
TPARSM	stores 3-D plot parameters in plot file Calls: PLOTB Called by: THRPLT	THREED	9
TPLOT	does plotting for 3-D or oblique view prog. Calls: CON,HIDDEN,MOVEKA,PLOT, PLOTS,PRFRD,THRPLT Called by: THREED	THREED	9

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
UNPLT	plots the unit symbol between two boundary points. Calls: PLOT,SZPLOT Called by: - *Not currently being loaded into executable programs	*FTMAIN *RTM	* 11 * 12
UTMPRT	convert floating point number to an NCHAR aplhanumeric array with Al format. Done to preserve precision in printing of large floating point numbers. Calls: nothing Called by: TITLE	CLOAD LLOAD PLOAD RLOAD TLOAD	1 2 3 4 5
UTM2LL	calc. geographic coordinates from universal transverse mercator (UTM) coordinates. Calls: nothing Called by: CNVTR,DRFTPR,DRFTTH, INCON,INLOS,INPRS,INRTM INTHRD,LMAIN,PLBPFT,PLSPFT	CLOAD FTMAIN LLOAD PLOAD PROFL RLOAD RTM TLOAD	1 11 2 3 6 4 12 5
UTM2MG	convert UTM value to mil grid value that is in character format. Calls: nothing Called by: PRINFT,PRINR	MAINFT	13
VPL2	makes all the plotting calls for Versatec plotting. Calls: FACTOR, MOVEAA,MOVEKA,WHERE, WNUMBER,WPLOTS,WSYMBOL,XPLOT Swapped to by: V2SWP	VPL2	10,14

FILE	FUNCTION	EXECUTABLE PROGRAM(S)	DIAGRAM
VPRMS	sets the Versatec scales, window limits and stripping factor. Calls: nothing Called by: PLTRN	PLTRN	10
V2SWP	swaps to the Versatec plotting program VPL2. Calls: MOVEAA Called by: PLTRN	PLTRN	10
WINDOW	puts user defined area into core memory for faster execution. Calls: nothing Called by: PTMN	PROFL	6

IIC. EXPANDED PROGRAM DESCRIPTIONS

ADDREC This subroutine serves two purposes.

1. Add feature data records to an existing feature data file.
2. Change location information of any record already in the feature data file.

Location information gives the location of a point by specifying its latitude and longitude or its northing-easting values. The user can input location information by keying-in or by the digitizing tablet (not yet implemented). (Drawing 13)

SUBROUTINE ADDREC(IBLK,IWD,IFLAG)

Parameters

IBLK - Block number wheere feature data record is found.

IWD - Start position of record in block.

IFLAG - 0 = adding new record
1 = changing old record

AGAIN asks the user if he wants to do another plot of the same type. (Drawings 1,2,3,4,5)

SUBROUTINE AGAIN(IJ,IK)

Parameters

IJ - Flag denoting type of plot:

1	-	Line-of-Sight
2	-	Radar Terrain Mask
3	-	Contour
4	-	Axonometric
5	-	Perspective
6	-	3-Dimensional

IK - Flag denoting answer:

0	-	No
1	-	Yes

ALT This subroutine determines the elevation for a given location specified by the latitude and longitude. It reads the polynomial data base. (See subroutine NALT for gridded data base.) (Drawing 6)

SUBROUTINE ALT(X,Y,Z)

Parameters

X	-	Longitude of location in floating point seconds.
Y	-	Latitude of location in floating point seconds.
Z	-	Elevation above (X,Y) location in meters.

CALCPR Given the X,Y coordinates of a location (point) in a perspective plot, CALCPR calculates which profile the point is on and the point number along that profile.
(Drawing 11)

SUBROUTINE CALCPR(RY,RX,ISCAN,NPT)

Parameters

RY - Latitude or northing of location
RX - Longitude or easting of location
ISCAN - Profile number for location
NPT - Point number for location

CALCTH Given the X,Y coordinates of a location (point) in a 3-dimensional plot, CALCTH calculates which profile the point is on and the point number along that profile.
(Drawing 11)

SUBROUTINE CALCTH(RY,RX,ISCAN,NPT)

Parameters

RY - Latitude or northing of location
RX - Longitude or easting of location
ISCAN - Profile number for location
NPT - Point number for location

CHANGE This subroutine allows the user to change the following information in a feature data record:

1. The record code (numeric).
2. The record description (20 characters or less).
3. Location information (i.e. the latitude and longitude or northing-easting values of a point.) Location information is changed by calling subroutine ADDREC.

(Drawing 13)

SUBROUTINE CHANGE(IBLK,IWD)

Parameters

IBLK - Block # to read containing feature record

IWD - Word # where record starts

CMAIN is the main program for the contour option. CMAIN calls the subroutines which generate and display the contour map. At the option of the user, feature data points can be drawn on the plot. (See subroutine DRIVFT.)

User inputs are made in subroutine INCON.

(Drawing 1)

Parameters - None

CNVTR is the subroutine which controls the conversion of units between the WGS and UTM system.
(Drawing 6)

SUBROUTINE CNVTR(IFLAG,ISPHR,IH,IZONE,XIN,YIN,XOUT,YOUT,IZOUT)
Parameters

IFLAG = 1 for WGS to UTM (calls LL2UTM)
2 for UTM to WGS (calls UTM2LL)

ISPHR,IH,IZONE,XIN,YIN - See PARAM Common Area description.

If WGS to UTM, outputs are XOUT,YOUT,IZOUT, (the northing, easting, and zone).

If UTM to WGS, outputs are XOUT, and YOUT, (the latitude and longitude).

CON This subroutine plots the contour levels between two profiles. This version of CON is used to plot contours on contour and radar terrain mask plots. (See subroutine VCON for perspective and 3-D.)
(Drawings 1,8)

SUBROUTINE CON(IX,IY,IZ,NPTS)
Parameters

IX and IY are 256 x 2 arrays of X and Y coordinates.
IZ is a 256 x 2 array containing the Z coordinates for a profile. (This array is obtained from subroutine PRFRD (for contour plots) or subroutine JPRFRD (for RTM plots.)
NPTS is the number of points along the profile.

CONPLT does the initial plotting for the contour plot option. The title, plot parameters, and contour levels are written around the 8 x 8 inch plot in the middle of the Tektronix screen. CONPLT also calculates the X and Y scale factors to fit the plot area to the 8 inch square.
(Drawing 1)

SUBROUTINE CONPLT(XSC,YSC)

Parameters

XSC and YSC are the horizontal and vertical plot scales output in inches/terrain point.

CONTBD plots boundary type feature data on the contour plot option.
(Drawing 12)

SUBROUTINE CONTBD(RY,RX,ICODE)

Parameters

RY - Latitude or northing of boundary point
RX - Longitude or easting of boundary point
ICODE - Boundary type code

CONTFT plots single point feature data on the contour plot option. Each point inside the plot boundaries is plotted along with its appropriate military symbol. (See subroutine SPSYM).
(Drawing 12)

SUBROUTINE CONTFT(RY,RX,ICODE)

Parameters

RY - Latitude or northing of feature point
RX - Longitude or easting of feature point
ICODE - Code for single point feature. (See subroutine SYPSM for a description of ICODE).

COPSM records the number of hard copies that has been requested of a plot. Subroutine COPSM calls subroutine PLOTB which stores NC in an array; when the array is full, it is written to the plot file.
(Drawings 1,2,7,8,9)

SUBROUTINE COP(NC)
Parameter
NC - The requested number of plot copies.

COPY produces the requested number of hard copies of the Tektronix screen by sending the appropriate Tektronix command sequence.
(Drawings 10,16)

SUBROUTINE COPY(NC)
Parameter
NC - The requested number of plot copies.

CPARSM stores parameters in the plot file needed to do a contour plot to scale.
(Drawing 1)

SUBROUTINE CPARSM(XL,YL,XD,YD)
Parameters
XD - Difference between eastern and western boundaries in WGS or mil grid units.
YD - Difference between northern and southern boundaries in WGS or mil grid units.

XL and YL give these same differences multiplied by an appropriate scale factor. (See subroutine CONPLT and also the CONBLK Common Area).

CPARSM calls PLOTB which stores these parameters in the IPBUF array; when this array is full, they are written to the plot file.

CPLOT does the plotting for the contour map program, as follows:

1. It calls subroutine CONPLT which writes the title and plot parameters.
2. For each profile, it obtains the elevation values by calling subroutine PRFRD and plots the contour levels by calling subroutine CON.
3. If feature data is to be plotted, CPLOT computes the values of the variables in the CONTSC Common Area.

(Drawing 1)

SUBROUTINE CPLOT

Parameters - none

DASH plots a dashed line from the present screen position to the point (XZ,YZ).

(Drawing 10)

SUBROUTINE DASH(XZ,YZ,NUMDPI)

Parameters

XZ - X coordinate of point in inches.

YZ - Y coordinate of point in inches.

NUMDPI - Number of dashes per inch.

DMSSEC converts a latitude or longitude (in degrees, minutes, seconds and direction) to signed seconds.
(Drawings 1,2,3,4,5,13)

SUBROUTINE DMSSEC(LDMS,XL)
Parameters

LDMS - Latitude or longitude in degrees, minutes, seconds, and direction (See CONBLK,LOSBLK,PRSBLK,RTMBLK, and THRBLK Common Areas for description of the variable LDMS).

XL - Latitude or longitude in signed seconds.

DOPARM requests the user to enter a 4 character name for the file which contains the plot parameters. It then calls PLOTPM which prints the parameters on the terminal, the Versatec printer or both.
(No drawing)

Parameters - none (This is a main program.)

DRFTPR is the driver to plot feature data on perspective plots. It computes the scale factors SCX and SCY. (See SCOM Common Area).
DRFTPR then calls subroutine FTOPEN which opens the feature data file. It then reads feature data from this file (single points and boundary points), computes the X and Y coordinates of these points and calls the appropriate plotting subroutines. (See PRSBD and PRSFT).
(Drawing 11)

SUBROUTINE DRFTPR
Parameters - none

DRFTTH is the driver to plot feature data on 3-dimensional plots. It calls subroutine FTOPEN which opens the feature data file. It then reads feature data from this file (single points and boundary points), computes the X and Y coordinates of these points, and calls the appropriate plotting subroutine. (See THRBD and THRFT).
(Drawing 11)

SUBROUTINE DRFTTH
Parameters - none

DRIVFT is the driver subroutine for plotting feature data on a contour plot or radar terrain mask plot.

DRIVFT opens the feature data file and calls subroutine RDHDFT to read the header record (block 0 of the file). It then reads records from the file and calls the appropriate routines for plotting feature data, as follows:

Subroutine PLSPFT	single point feature data
Subroutine PLBPFT	boundary type feature data

(Drawings 1,8,12)

SUBROUTINE DRIVFT
Parameters - none

DSHSM stores parametes needed to draw a dashed line from one position to another. This version of DASH is used for the line-of-sight plot. (It is called by subroutine LOSPLT).
(Drawing 2)

SUBROUTINE DASH(X,Y,NDPI)
Parameters

X - X coordinate of point where line will end.
Y - Y coordinate of point where line will end.
NDPI - Number of dashes per inch

ERRFT This subroutine is used by the programs which create and modify the feature data files. It reports an error code if an error occurs in opening, closing, writing to or reading from a file.
(Drawing 13)

SUBROUTINE ERRFT(IERR,K)
Parameters

IERR - FORTRAN error code (from RDBLK)
K - Integer flag indicates which call error occurred on.

ERTOPT This subroutine reads the elevation correction option entered from the keyboard. The user can choose option number

1. Earth curvature and atmospheric refractivity correction.
2. Earth curvature correction.
3. No correction.

Based on the above option, ERTOPT computes the value of COEF, the correction to be applied, and stores this value in the EARTH Common Area. (These correction options can be used with all plot types except the contour plot).
(Drawings 2,3,4,5)

SUBROUTINE ERTOPT
Parameters - none

ERTPLT plots the earth correction option chosen by the user in subroutine ERTOPT. Nothing is written for the 'no correction' option.
(Drawings 2,8)

SUBROUTINE ERTPLT(XP,YP)
Parameters
XP and YP are the X and Y coordinates on the Tektronix screen where writing will start.

ERTPRT prints on the Tektronix screen (Unit number IWRT) the earth correction option chosen by the user in subroutine ERTOPT. Nothing is printed for the 'no correction' option.
(Drawings 2,3,4,5)

SUBROUTINE ERTPRT
Parameters - none

FTMAIN is the driver program for plotting feature data on a perspective or 3-dimensional plot. ("FTMAIN.SV" is swapped to from subroutine FTSWP). Note that FTMAIN is used for drawing grids on these two plot types as well as feature data.

Before plotting is done, FTMAIN reads necessary information from the disk file "PNTFL" and opens the plot file "PLOTF". It then calls the appropriate subroutines for plotting a grid and/or feature data.
(Drawings 7,9,11)

Parameters - none (This is a main program)

FTOPEN opens the feature data file, reads the first block, and extracts from the header record the number of feature records and a flag variable used to specify the conversion of units between the WGS and UTM system. (See PLOTVR Common Area). (Drawing 11)

SUBROUTINE FTOPEN(IOPEN,IBUF)

Parameters

IOPEN = 1 if the file is not successfully opened and read.

IBUF = returns the header record

FTSWP fills the file "PNTFL" with the appropriate Common blocks and swaps to "FTMAIN.SV" to do the feature data plotting and grid line plotting.

FTSWP fills the "PNTFL" file as follows: values from various Common blocks are put into the IBUF array (See BUF Common Area) and this array is written to block 0 of the disk file. The plot buffer, IPBUF is written to block 1 and the hidden point matrix, MHIDE, to following blocks. (See also PLTBF and HIDE Common Areas).
(Drawings, 7,9)

SUBROUTINE FTSWP

Parameters - none

GRDPRS plots grid lines on the perspective plot. Spacing between grid lines is specified by the variable IGRID (See GRDBLK Common Area).
Each line is plotted by calling PRSLIN, which in turn calls PTCALC and PLOT.
GRDPRS uses the point file created by PPLOT. (See the PTBLK and PTFIL Common Areas).
(Drawing 11)

SUBROUTINE GRDPRS
Parameters - none

GRDRTM draws grid lines on radar terrain mask plots. The grid consists of vertical and horizontal lines. Spacing between grid lines is specified by the variable IGRID. (See GRDBLK Common Area).
GDRTM computes grid spacing (in inches) and the intersection of the grid with the plot boundary, and calls PLOT to write this information to the plot file.
(Drawing 8)

SUBROUTINE GRDRTM
Parameters - none

GRDTHR draws grid lines on 3-dimensional plots. Spacing between grid lines is specified by the variable IGRID. (See GRDBLK Common Area).
GRDTHR uses the point file created by TPLOT. (See the PTBLK and PTFIL Common Areas). It calls PLOT to write this information to the plot file.
(Drawing 11)

SUBROUTINE GRDTHR
Parameters - none

HIDDEN sets, clears, or checks bits in the hidden point matrix (array MHIDE in Common HIDE).
(Drawings 7,9,11)

SUBROUTINE HIDDEN(BIT,IFLAG,ICODE)

Parameters

BIT - number of the point in the matrix to set, clear, or check.

ICODE - 0 = clear the bit
1 = set the bit to 1
2 = check the bit, return status in IFLAG

IFLAG - 0 = bit is not set
1 = bit is set

IAZCHK FUNCTION IAZCHK tests whether a periodic value is within a specified region.
(Drawings 6,8,11,12)

FUNCTION IAZCHK(AZ,AZ1,AZ2)

Parameters

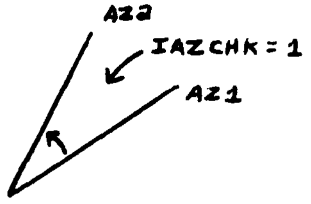
AZ - The test value

AZ1,AZ2 - Boundary values

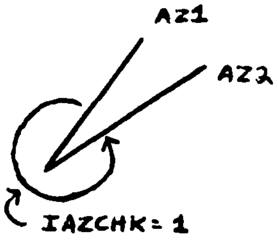
IAZCHK = 1 - Returned if within region
= 0 - If outside region

The following convention is used for determining whether an angular value is "within" a region:

AZ2 > AZ1



AZ1 > AZ2



ICHK FUNCTION ICHK tests whether a value is between two other values. (Drawings 11,12)

FUNCTION ICHK(A,A1,A2)

Parameters

A	-	The test value
A1,A2,	-	Boundary values
ICHK = 1	-	Returns if between boundary values
= 0	-	If outside boundary values

INCON is the subroutine where most of the user inputs for the contour plot are made. It reads these inputs from the terminal, checks their validity and prints them on the terminal in a readable format. INCON requests the following information from the user:

1. Plot boundaries.
   If data is being entered in the WGS system (INPUT = 1), these boundaries are entered as latitudes and longitudes in degrees, minutes, seconds, and direction, (LDMS). DMSSEC is called to convert these units to signed seconds. If data is being entered in the mil grid system (INPUT = 2), INCON calls MGBOUN to obtain the boundary inputs. (IBND).
2. Latitude and longitude intervals. (D). The user inputs these as latitude and longitude intervals in seconds (WGS system) or northing and easting values in meters (mil grid system).
3. The minimum contour level in meters. (ICMIN)
4. The contour interval in meters. (ICDEL)
5. The tic mark spacing. (ITIC) i.e., the number of terrain points between tic marks. A value of zero gives no tic marks.
6. The desired number of plot copies. (NCOPY)

INCON also computes the number of points along a profile (NPTS) and the number of profiles (NSCAN).

After printing the above inputs on the terminal, INCON fills the IPMBUF array with these values, and other relevant variables in the Common blocks and writes the array to block 0 of the parameters file (PF).

Finally, INCON checks and informs the user if none of the desired contour area is within the data base.

See also the CONBLK,IBND, and CONPRM Common Areas.
(Drawing 1)

SUBROUTINE INCON (IGD)

Parameters

IGD INCON sets IGD=1 if the data base does not contain any of the contour area; otherwise IGD = 0.

INLOS is the subroutine where most of the user inputs for the line-of-sight plot are made. It reads these inputs from the terminal, checks their validity and prints them on the terminal in a readable format.

INLOS requests the following information from the user:

1. Locations of endpoints (initial and terminal points). If data is being entered in the WGS system (INPUT = 1), these locations are latitudes and longitudes in degrees, minutes, seconds, and direction. (LDMS). Subroutine DMSSEC is called to convert these units to signed seconds. If data is input in the mil grid system (INPUT = 2), INLOS calls subroutine MGCORD to obtain the endpoint locations. (IA).

2. The number of profile points. (NPTS) NPTS must have a value from 2 to 750. If NPTS = 0, an approximate 100 meter spacing between points is used.

3. Earth correction option. INLOS calls ERTOPT to obtain this input.

4. Plot/Print option. (IPRT)

```
   IPRT = 0  plot only
          1  plot and table
          2  table only
```

5. The desired number of plot copies. (NCOPY)

After printing the above inputs on the terminal, INLOS fills the IPMBUF array with these values and other relevant variables in the Common blocks and writes the array to block 0 of the parameters file (PF).

Finally, INLOS checks and informs the user if the endpoints are outside the data base.

See also LOSBLK,LOCUTM, and CONPRM Common Areas.

(Drawing 2)

SUBROUTINE INLOS(IGD)

Parameters

IGD INLOS sets IGD = 1 if the line-of-sight endpoints are outside the data base; otherwise IGD = 0.

INPRS is the subroutine where most of the user inputs for the perspective plots are made. It reads these inputs from the terminal, checks their validity and prints them on the terminal in a readable format. INPRS requests the following information from the user:

1. Location of observer.
   If data is being entered in the WGS system (INPUT = 1), the location is entered as a longitude and latitude value in degrees, minutes, seconds and direction, (LDMS). DMSSEC is called to convert these units to signed seconds. If input data is in the mil grid system (INPUT = 2), INPRS calls MGCORD to obtain the observer location, (IA).
2. The plot bearing in degrees. (AZO)
3. Observer height in meters. (HT)
4. Total radial length in meters. (DST)
5. The spacing along the radial in meters. (DSTM)
6. The spacing between radials in degrees. (DAZ)
7. The field of view in degrees. (VA) A value of VA=0. causes a default value of 60 degrees to be used.
8. Plot options.
   INPRS calls PLTOPT to obtain these inputs. (See PLTOPT).
9. Vertical exaggeration factor. (VEX)
10. Earth correction option. This is obtained by calling ERTOPT. (See ERTOPT).

11. The desired number of plot copies. (NCOPY)

INPRS also computes the number of points along a radial (NPTS) and the number of radials (NSCAN).
After printing these inputs on the terminal, INPRS fills the IPMBUF array with these values and other relevant variables in the Common blocks and writes the array to block 0 of the parameters file (PF).
Finally, INPRS checks and informs the user if the observer position is outside the data base.
See also the PRSBLK,LOCUTM, and CONPRM Common Areas.
(Drawing 3)

SUBROUTINE INPRS(IGD)
Parameters

IGD INPRS sets IGD=1 if the observer location is not in the data base; otherwise IGD=0.

INRTM is the subroutine where most of the user inputs for the radar terrain mask plots are made. It reads the inputs from the terminal, checks their validity, and prints them on the terminal in a readable format.
INRTM requests the following information from the user:

1. The masking option. (IFLAG). Values of IFLAG and the corresponding options are:
   1. Safe area contours.
   2. Acquisition contours.
   3. Safe area below given ceiling.
   4. Fields of fire.
2. Cross hatching option. (ICH)
   The user specifies whether or not he wants cross hatching to appear on the plot.

3. Location of observer.
   If data is being entered in the WGS system (INPUT = 1), the location is entered as a longitude and latitude value in degrees, minutes, seconds, and direction, (LDMS). DMSSEC is called to convert these units to signed seconds. If input data is in the mil grid system (INPUT = 2), INTRM calls MGCORD to obtain the observer location, (IA).
4. Observer height in meters. (IH)
5. Ceiling height in meters. (IFH)
   (only if masking option 3 is chosen).
6. Coverage radius in kilometers. (RK)
7. Spacing along a radial in meters. (DR)
8. Bearing of the first and last radials (BI and BF, respectively). Both are in degrees.
9. Spacing between radials in degrees. (DB)
10. Minimum contour level in meters. (ICMIN)
11. Contour interval in meters. (ICDEL)
12. Grid spacing in seconds. (IGRID)
    IGRID = 0 indicates no grid will be drawn.
13. Earth curvature and atmospheric refractivity correction. This is input through a call to ERTOPT. (See ERTOPT).
14. The desired number of plot copies. (NCOPY)

INRTM also computes the number of radials (NR) and the number of points along a radial (NP).

After printing these inputs on the terminal, INRTM fills the IPMBUF array with these values and other relevant variables from the Common blocks and writes the array to block 0 of the parameters file (PF).

Finally, INRTM checks and informs the user if the observer position is outside the data base.

See also RTMBLK,LOCUTM and CONPRM Common Areas.

(Drawing 4)

SUBROUTINE INRTM(IGD)

Parameters

IGD   INRTM sets IGD = 1 if the observer position is outside the data base; otherwise IGD = 0.

INTHRD is the subroutine where most of the user inputs for the 3-dimensional plots are made. It reads these inputs from the terminal, checks their validity and prints them on the terminal in a readable format.

INTHRD requests the following information from the user:

1. Plot boundaries.

   If data is being entered in the WGS system (INPUT = 1), these boundaries are entered as latitudes and longitudes in degrees, minutes, seconds and direction. (LDMS). DMSSEC is called to convert these units to signed seconds. If data is being entered in the mil grid system (INPUT = 2), INTHRD calls MGBOUN to obtain the boundary inputs, (IBND).

2. Latitude and longitude spacing. (D). The user can input these as latitude and longitude intervals in seconds (WGS system), or northing and easting values in meters (mil grid system).

3. View Face. (JV)

   This is the boundary closest to the projection plane.

4. Direction from which area is viewed. (JA)

5. Reference elevation in meters. (IZMIN)

6. Plot options.

   INTHRD calls PLTOPT to obtain these inputs. (See PLTOPT).

7. Vertical exaggeration factor. (VEX)

8. Earth correction option.
   INTHRD calls ERTOPT to obtain this input. (See ERTOPT).

9. Desired number of plot copies. (NCOPY)

After printing these inputs on the terminal, INTHRD fills the IPMBUF array with these values and other relevant variables in the Common blocks and writes the array to block 0 of the parameters file (PF). Finally, INTHRD checks and informs the user if none of the plot area is inside the data base.
See also the THRBLK,IBND, and CONPRM Common Areas.
(Drawing 5)

SUBROUTINE INTHRD(IGD)
Parameters

IGD	INTHRD sets IGD = 1 if none of the plot area is inside the data base; otherwise, IGD = 0.

INVM Given two geographic endpoints in signed seconds, INVM computes tne azimuth in radians and the geodetic distance between the two points in meters.

This subroutine is used by the line-of-sight plot main program, LMAIN, for computing the azimuth and distance between the initial and terminal points. (SODINV is also called in the course of this computation).
(Drawing 2)

SUBROUTINE INVM(X1,X2,X3)

DIMENSION X1(2,2)

Parameters

X1(1,1) - Latitude of first endpoint in signed seconds.
X1(2,1) - Longitude of first endpoint.
X1(1,2) - Latitude of second endpoint.
X1(2,2) - Longitude of second endpoint.
X2 is the azimuth in radians
X3 is the geodetic distance in meters.

X1,X2, and X3 are all double precision variables.

IUNPCK This subroutine unpacks a computer word into two integers. The two integers receive the respective bytes (8 bits).
(Drawings 1,2,3,4,5,6,15)

SUBROUTINE IUNPCK(IA,IX,IY)

Parameters

IA	-	Packed computer word.
IX	-	Integer coming from the left half of IA.
IY	-	Integer coming from the right half of IA.

KAM2AS This subroutine translates an alphanumeric array in A2 format into an array of ASCII decimal equivalent (ADE) integers. This is used in the output to a Tektronix terminal.
(Drawing 15)

SUBROUTINE KAM2AS(LEN,IAM,IADE)

Parameters

LEN	-	Number of characters to be translated.
IAM	-	Alphanumeric array in A2 format.
IADE	-	Array of ADE integers.

LL2UTM This subroutine converts from latitude-longitude units to UTM units (northing, easting and zone).

(Drawings 1,2,3,4,5,6,11,12)

```
SUBROUTINE LL2UTM(L,XLTS,XLNGS,XNOR1,XEAST1,IZONE1)
```

Parameters

L is an integer from 1 to 8 used to specify which type of spheroid constants is being used for the conversion. (See PARAM Common Area).

XLTS	-	Latitude in signed seconds.
XLNGS	-	Longitude in signed seconds.
XNOR1	-	Northing value in meters.
XEAST1	-	Easting value in meters.
IZONE1	-	UTM zone.

LMAIN is the main program for the line-of-sight plot. LMAIN calls subroutines which generate a profile of the terrain between two geographic points. The completed profile may be displayed on the Tektronix terminal as a plot and/or a table of elevation values.

At the option of the user, corrections for the earth's curvature and/or atmospheric refractivity may be made. User inputs for the line-of-sight plot are made in subroutine INLOS. (See also subroutine ERTOPT.)

(Drawing 2)

Parameters - none

LOSPLT does the plotting for the line-of-sight plot. It produces a graph of elevation in meters versus distance in kilometers along the line-of-sight from the initial to the terminal point.

LOSPLT first determines the origin ana plot size (See also subroutine LPARSM), and computes appropriate scale factors. Then it writes a title on the plot, draws and labels the axes, and plots the elevations.

LOSPLT obtains the elevations from the IB array. (See the POINT Common Area).
(Drawing 2)

SUBROUTINE LOSPLT
Parameters - none

LOSPRT This subroutine prints on the Tektronix terminal a table of the elevation values along the line between the initial and terminal points.

LOSPRT obtains the elevations from the IB array. (See the POINT Common Area).
(Drawing 2)

SUBROUTINE LOSPRT
Parameters - none

LPARSM stores parameters in the plot file needed to draw a line of sight plot to scale.
(Drawing 2)

SUBROUTINE LPARSM(XO,YO,XLNGTH,YLNGTH)

Parameters

XO and YO give the plot origin in inches.

XLNGTH and YLNGTH give the length and height of the plot in inches.

Subroutine LOSPLT assigns values to these variables before calling LPARSM. (See subroutine LOSPLT and the PLTSAV Common Area).

LPARSM calls PLOTB which stores these parameters in the IPBUF array; when this array is full, they are written to the file.

MAINFT This is the main program for creating and modifying feature data files. MAINFT calls subroutines which enable the user to add, delete, change, or print records in a feature data file. MAINFT requests the user to supply the name of the feature data file. If it is a new file, the user also needs to give a file description (see HEADER Common Area) and choose whether input is to be in the WGS or UTM system. If it is an old file, MAINFT verifies that it is a feature data file before any changes are made.
After printing on the terminal the number of records (boundary and single point), the user is asked whether input will be by digitizing from maps or by typing in. If input will be by digitizing, MAINFT computes the map scales using four reference points. (See REFVAL Common Area).

MAINFT then calls the appropriate subroutines for modifying the file. (See also subroutines ADDREC,CHANGE,PRINFT, and SRCHFI.)
(Drawing 13)

Parameters - none

MGBOUN This subroutine allows the user to input values for plot boundaries in mil grid form. MGBOUN is used for the contour and 3-dimensional plot types.

MGBOUN requests the following information from the user:

1. Plot boundaries. (IBND)
   Each boundary is input in mil grid form as one 4-digit number (northing or easting value). (See the IBND Common Area).
2. The zone row. (IZR)
3. The zone column. (IZC)

(Drawings 1,5)

SUBROUTINE MGBOUN(ITYPE,BOUND,XBOUND)

Parameters

ITYPE specifies which boundary as follows:

ITYPE Value	Boundary
1	Southern
2	Northern
3	Western
4	Eastern

BOUND Takes on X value for Western/Eastern Boundary
Takes on Y value for Southern/Northern Boundary

XBOUND Takes on X value for Southern/Northern Boundary
Takes on Y value for Western/Eastern Boundary

MGCORD allows the user to input a UTM value in mil grid form. This value is either an observer location (perspective and radar terrain mask plots) or an initial or terminal point (line-of-sight plot).

MGCORD requests the user to enter the UTM value as an 8-digit number of form 'EEEENNNN'.

(Drawings 2,3,4,13)

SUBROUTINE MGCORD(ITYPE,XEAST,XNORTH)

Parameters

ITYPE Specifies whether the UTM value is an observer location or an initial or terminal point as follows:

ITYPE VALUE	Location
1	Observer location
2	Initial point
3	Terminal point
4	Feature data point

XEAST Easting coordinate

XNORTH Northing coordinate

MGSET This subroutine sets up certain parameters for the use of UTM coordinates. Subroutine SETUP calls MGSET if the user chooses to input data in mil grid units.

MGSET requests the user to input the following:

1. The smallest possible UTM easting value for the lower left mil grid.
2. The smallest possible UTM northing value for the lower left mil grid.
3. The number of mil grid rows.
4. The number of mil grid columns.

Based on these inputs, MGSET computes prefix values for northing and easting and minimum northing and easting values. (See also the MGBLK and ZONEZ Common Areas).
(Drawings 1,2,3,4,5,13)

SUBROUTINE MGSET
Parameters - none

MOVEAA This subroutine moves a specified number of elements from one array to another. It is a utility type routine used by many of the programs. For example, several subroutines use MOVEAA to move the values of variables from Common blocks into a buffer array which is then written onto a disk file.
(Drawings - all except 15, and 16)

SUBROUTINE MOVEAA(NUM,IFROM,ITO)
Parameters

NUM - The number of values to be moved.
IFROM - Array from which the transfer will be made.
ITO - Array to which the transfer will be made.

MOVEKA This subroutine puts a value, IVAL, into a specified number of elements of an array. It is a utility type routine used by many programs.
(Drawings 1,2,6,7,8,9,10,11,12,13,14)

SUBROUTINE MOVEKA(NUM,IVAL,ITO)

Parameters

NUM - The number of values to be moved.

IVAL - The value which will be put into the ITO array.

ITO - Array to which the transfer will be made.

NALT This subroutine determines the elevation for a given location specified by the latitude and longitude. It reads the gridded data base. (Note - see ALT for polynomial data base.)
(Drawing 6)

SUBROUTINE NALT(X,Y,Z)

Parameters

X - Longitude of location in floating point seconds

Y - Latitude of location in floating point seconds

Z - Elevation above (X,Y) in meters.

NPAGE is a utility subroutine which notifies the user to make a hard copy of information currently appearing on the Tektronix terminal, if desired, and then to erase the screen. (Drawings 1,2,3,4,5,7,8,9,10)

SUBROUTINE NPAGE
Parameters - none

ONEPNT This is to be the digitizing subroutine for obtaining feature data points from maps. (It is not yet fully implemented).

(Drawing 13)

SUBROUTINE ONEPNT(IX,IY)
Parameters
IX - X coordinate of cursor
IY - Y coordinate of cursor

ONEPNT will be used by MAINFT and ADDREC in the process of creating and modifying feature data files. It calls subroutine THMBPT to obtain the cursor coordinates.

PERSP is a main program which calls the subroutines involved in the actual plotting for the perspective plot. It is swapped to from PMAIN. (See also PRSWP).

Before plotting is done, PERSP reads necessary information from the diskfile "SWPFL" and the profile file. It then calls PPLOT to perform plotting and FTSWP if feature data is to be plotted.

(Drawings 3,7)

Parameters - none

PLBPFT This subroutine is called by DRIVFT and is used in plotting boundary type feature data on a contour or radar terrain mask plot.

PLBPFT calculates the position of each boundary point, converts units if necessary, and calls the appropriate boundary plotting routine. Which routine is called is determined by the variable IPLOT, as follows:

IPLOT Value	Routine Called
1	CONTBD
2	RTMBD

(See also the PLOTVR Common Area).

(Drawing 12)

SUBROUTINE PLBPFT

Parameters - none

PLOTB This subroutine is used to store plot information in an array, IPBUF. When the array becomes full, it is written to the plot file. (See the PLTBF Common Area).
(Drawings 1,2,7,8,9,11,12)

SUBROUTINE PLOTB(NUM,IBUF)

Parameters

NUM - The number of elements to be added to the IPBUF array

IBUF - Array containing the elements being added

PLOTPM prints out the parameters in the "XXXX.PM" file that corresponds to the plot file "XXXX.PL" where XXXX is the 4-character name of the plot file.
PLOTPM obtains these plot parameters from block 0 of the parameter file. These parameters are mostly those input by the user in subroutine INCON,INLOS,INPRS,INRTM, or INTHRD.
Output can go either to the Tektronix terminal or the printer.
(Drawing 10)

SUBROUTINE PLOTPM

Parameters - none

PLSPFT This subroutine is called by DRIVFT and is used in plotting single point type feature data on a contour or radar terrain mask plot.

PLSPFT calculates the position of each point, converts units if necessary, and calls the appropriate single point plotting routine. Which routine is called is determined by the variable IPLOT, as follows:

IPLOT Value	Routine Called
1	CONTFT
2	RTMFT

(See also the PLOTVR Common Area).
(Drawing 12)

SUBROUTINE PLSPFT
Parameters - none

PLTOPT This subroutine requests the user to input from the terminal various options for the perspective and 3-dimensional plots.

These options are specified by the variable ICR as follows:

0. Grid lines.
1. Range lines.
2. Contour levels.
3. Range lines with grid lines.
4. Contour levels with grid lines.

Depending on the choice of options, the user is then requested to input one or more of the following:

1. Minimum contour level in meters (ICMIN) and contour interval in meters (ICDEL).
2. The grid spacing in seconds (WGS system, INPUT=1) or in meters (mil grid system, INPUT=2). (IGRID)
3. Whether all ridge lines are to be plotted (IRDGE).

These variables are stored in the GRDBLK,RIDGE, and IPLOPT Common Areas.

(Drawings 3,5)

SUBROUTINE PLTOPT(ICR)

Parameters

ICR - As listed above.

PLTPRT prints plot options chosen by the user on the Tektronix terminal. These options apply to all plot types except line-of-sight.
(Drawings 1,3,4,5)

SUBROUTINE PLTPRT(ICR)

Parameter

ICR (See subroutine PLTOPT for list of ICR values and corresponding plot options).

Depending on the value of ICR, PLTPRT prints contour levels, grid spacing, and whether range lines will be plotted. Inputs relevant to these quantities are made in PLTOPT (for the perspective and 3-dimensional plots), INCON (contour plot) or INRTM (radar terrain mask plot).

PLTRN Is the main program which calls the routines which display a plot file on the Tektronix terminal or Versatec plotter.
If PLTRN is swapped to from one of the plotting programs by subroutine PSWAP, the temporary plot file, "PLOTF", is displayed on the Tektronix terminal.
If PLTRN is run as a stand alone program, the permanent plot file, "XXXX.PL", is displayed on the Tektronix terminal, Versatec plotter, or both, as specified by the user.
(Drawings 1,2,7,8,9,10)

Parameters - none

PLTSV gives the user the option of saving a plot output file for later use. If the user does not want to save the file PLTSV deletes the temporary plot file, "PLOTF" and corresponding parameters file, "PLPARM".

If the plot file is to be saved, PLTSV requests the user to supply a 4-character name for the permanent plot file. This file is "XXXX.PL" and the corresponding parameters file is "XXXX.PM", where XXXX is the user supplied name.

(Drawings 1,2,3,4,5)

SUBROUTINE PLTSV

Parameters - none

PL2 calls the subroutines which are involved in displaying a plot file on the Tektronix terminal. It is called by PLTRN.

(Drawing 10)

SUBROUTINE PL2

Parameters - none

PMAIN is the main program for the perspective plot. PMAIN calls subroutines which generate a perspective view, i.e. what an observer would see from a given location. The user may choose from the following plot options:

Grid lines
Range lines
Contour levels
Range lines with grid lines
Contour levels with grid lines.

At the option of the user, corrections for the earth's curvature and/or atmospheric refractivity may be made.

Feature data points can also be plotted on the perspective view.

User inputs for the perspective plot are made in subroutine INPRS. (See also subroutines ERTOPT,FTMAIN, and PLTOPT.)

(Drawing 3)

Parameters - none

PMPRT This subroutine adds the name of the profile file being used for the plot to the disk file on Unit #44 which contains other plot parameters. These parameters can be printed by PLOTPM. (See DOPARM and PLOTPM.)
The profile file name is stored in the PRFIL Common Area.
(Drawings 1,2,3,4,5)

SUBROUTINE PMPRT
Parameters - none

PPARSM stores parameters needed to draw a perspective plot to scale.
(Drawing 7)

SUBROUTINE PPARSM(XLEN,YLEN)
Parameters
XLEN and YLEN give the length and height of the plot in inches.

PRSPLT assigns values to these variables before calling PPARSM. PPARSM calls PLOTB which stores XLEN and YLEN in the IPBUF array; when this array is full, they are written to the plot file.

PPLOT does the plotting for the perspective plot as follows:

1. It calls PRSPLT which computes plot parameters and writes the plot title.
2. For each profile, it obtains the elevations of points along the profile by calling JPRFRD, which reads this information from the profile file.

Based on this information, PPLOT computes the coordinates of each point along the profile. Points which are hidden by terrain features are also computed so that these will not be drawn on the plot. (See also the HIDCM Common Area).

If feature data or a grid is to be drawn on the plot, PPLOT writes the coordinates of the points to the Points file, for later use by the subroutines which plot the grid or feature data.

(Drawing 7)

SUBROUTINE PPLOT

Parameters - none

PRCEED asks the user if he wants to proceed with the inputs he has entered. Each main program (CMAIN,PMAIN, etc.) calls PRCEED immediately after user inputs have been made in INCON,INLOS,INPRS,INRTM, or INTHRD.

If the user chooses not to proceed, the main program stops and the user can run it again with different inputs.

(Drawings 1,2,3,4,5)

SUBROUTINE PRCEED(IK)

Parameter

IK - Flag denoting user answer

0	-	proceed
1	-	do not proceed.

PRFRD This subroutine retrieves the previously computed elevation values of points along a profile, when these values are needed by the plotting programs.

PRFRD reads the elevations from the profile file. (See also PRFWRT which writes this information to the disk file). After being read from disk, the elevation values are put into the array IA. (See also the POINT Common Area).
PRFRD is used for the contour, line-of-sight and 3-dimensional plots.
(Drawings 1,2,9)

SUBROUTINE PRFRD(NPTS,IA)
Parameters

NPTS	-	Number of points in profile
IA	-	Array to receive the elevations

JPRFRD is almost identical to PRFRD. It has the same function as PRFRD and is the version used for the perspective and radar terrain mask plots.
(Drawings 7,8)

JPRFRT This subroutine retrieves the terrain profiles needed by the main programs before plotting can be done. The profiles are either created by swapping to "PROFL.SV" or obtained from a previously created file.

The user is asked if he wants to use an existing profile file. If so, JPRFRT opens the existing file, XXXX.PF, and checks the file's validity. If the file is not valid, control returns to the beginning of JPRFRT and the user may try another file; if valid, JPRFRT returns control to the main program.

If the user chooses to create a new profile file, JPRFRT requests a file name (See PRFIL Common Area), and swaps to "PROFL.SV" which computes the profiles. Before swapping, the ICOM array is filled with necessary variables from the Common blocks and is written to block 0 of the disk file "SWPFL". This information is subsequently read from the disk and used by PROFL.

(Drawings 2,3,4,5)

SUBROUTINE PRFRT

Parameters - none

WPRFRT serves the same purpose as JPRFRT. It is a slightly different version of the same subroutine.

CMAIN uses WPRFRT and the other main programs use JPRFRT.

(Drawing 1)

PRFWRT This subroutine writes the elevations of points along a profile into the profile file. This information is retrieved later by subroutine PRFRD or JPRFRD which read from this file. PRFWRT obtains the elevations from an array, IA. (See also the POINT Common Area). The elevation values are first put into a buffer array, IPBUF, which is written to a block of the disk file. This use of the buffer array reduces the number of writes necessary.
(Drawing 6)

SUBROUTINE PRFWRT(NPTS,IA,KK)
Parameters

NPTS - Number of points in the elevation array.

IA - The array containing elevations of the points along the profile.

KK - A flag variable which indicates the last time to write the buffer array to the disk file.

PRINFT prints the contents of a feature data file in a readable format. The contents are printed on the Tektronix terminal one screen at a time, giving the user the opportunity to get a hardcopy of a particular screen. (See MAINFT,ADDREC and CHANGE for the creation and modification of feature data files).
(Drawing 13)

SUBROUTINE PRINFT
Parameters - none

PRINR prints on the Tektronix terminal the contents of one record in a feature data file. (A feature data file contains single point and boundary records. See also ADDREC and CHANGE). Boundary records contain more than one point. PRINR prints the location of these points in degrees, minutes, and seconds (WGS system) or as a mil grid value (UTM system).
(Drawing 13)

SUBROUTINE PRINR(INRPTS)

Parameters

INRPTS - The number of points used for the boundary.

PROFL is the main program involved in computing terrain profiles. It is swapped to from the main programs, CMAIN,LMAIN, etc. (See also JPRFRT and WPRFRT).
Before profiles are computed, PROFL reads block 0 of the disk file "SWPFL" (Swap file), and puts this information into the Common blocks. It then calls PTMN to get the profiles.
The profiles are written to a disk file for later use. (See PTMN,PRFWRT,PRFRD, and JPRFRD, the subroutines involved in reading from and writing to this file).
(Drawings 1,2,3,4,5,6)

Parameters - none

PRSBD This subroutine plots boundary type feature data on a perspective plot.
(Drawing 11)

SUBROUTINE PRSBD(RY,RX,ICODE)

Parameters

RY - Latitude or northing of boundary point.

RX - Longitude or easting of boundary point.

ICODE - Boundary type code. (See SPSYM for a description of ICODE).

PRSFT plots single point feature data on a perspective plot. Each point not hidden by terrain features is plotted along with its appropriate military symbol. (See SPSYM).
(Drawing 11)

SUBROUTINE PRSFT(RY,RX,ICODE)

Parameters

RY - Latitude or northing of feature point.

RX - Longtidue or easting of feature point.

ICODE - Code for single point feature. (See SPSYM for a description of ICODE).

PRSLIN plots a feature line between 2 positions on a perspective plot. PRSLIN is called by GRDPRS which draws grid lines on a perspective plot and by PRSBD which draws boundary type feature data on the plot.
(Drawing 11)

SUBROUTINE PRSLIN(IC,NC,P)

Parameters

IC - Two dimensional array containing the profile numbers for the 2 positions.

NC - Two dimensional array containing the point numbers for the 2 positions.

P - Array containing the locations of the 2 positions.

P(1,1)	-	X coordinate
P(2,1)	-	Y coordinate
P(1,2)	-	X coordinate
P(2,2)	-	Y coordinate

(See also the PRPT Common Area).

PRSPLT does the initial plotting for the perspective plot option. It prints a title on the plot and calculates the plot length and width in inches. The length and width are stored in a disk file for later use. (See also PPARSM and PLOTB).
(Drawing 7)

SUBROUTINE PRSPLT(PSI,XLT)

Parameters

PSI - Angle from vertical to center of projection (in radians).

XLT - Distance from observer to center of projection plane (in inches).

PRSWP PMAIN uses PRSWP for swapping to "PERSP.SV" where the perspective view plotting is done.
Before swapping, PRSWP fills the ICOM array (see the SWPCM Common Area) with variables from relevant Common blocks and writes the array to block 0 of the disk file "SWPFL". This information is subsequently read from the disk by PERSP and used in drawing the plot.
(Drawing 3)

SUBROUTINE PRSWP

Parameters - none

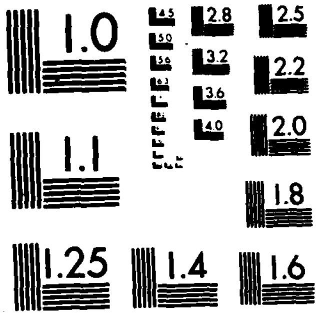
PSWAP This subroutine is used by all five plot types to swap to "PLTRN.SV" which contains the routines for displaying a plot file on the Tektronix terminal or Versatec plotter. Before swapping, PSWAP fills the ICOM array with variables from the Common blocks and writes this information to block 0 of the plot file. It also writes a final block to this file. (Drawings 1,2,7,8,9,11)

SUBROUTINE PSWAP
Parameters - none

PTCALC calculates the screen coordinates of a location (point) specified by which profile the point is on and the point number along that profile. It is used by the subroutines whicn plot feature data on a perspective or 3-dimensional plot. PTCALC uses the points array computed previously by PPLOT or TPLOT to obtain these coordinates. (See also the PTBLK Common Area).
(Drawing 11)

SUBROUTINE PTCALC(ISCAN,NPT,X,Y)
Parameters
ISCAN - Profile number for location.
NPT - Point number for location.
X,Y - Screen coordinates of location.

AD-A137 977 FEED (FIELD EXPLOITATION OF ELEVATION DATA) SOFTWARE 2/3
DOCUMENTATION(U) GEORGIA INST OF TECH ATLANTA
ENGINEERING EXPERIMENT STATION J G JAY ET AL. AUG 83
UNCLASSIFIED ETL-0335 DAAK70-82-K-0204 F/G 9/2 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

PTMN This subroutine gets the profiles for PROFL. For each profile PTMN calls PTSDR to obtain the elevations of points along the profile and PRFWRT to write these elevations into the profile file.

For each of the five plot types, PTMN computes the correct inputs to PTSDR and, when necessary, calls CNVTR for the proper conversion of units.

(Drawing 6)

SUBROUTINE PTMN

Parameters - none

PTRD This subroutine retrieves information previously computed by PPLOT or TPLOT for use by the subroutines which plot a grid or feature data on a perspective or 3-dimensional plot.

This information consists of the coordinates of points along a profile and is obtained by PTRD from the Points file. (See also the PTBLK Common Area).

(Drawing 11)

SUBROUTINE PTRD

Parameters - none

PTS This subroutine obtains elevations of points along a profile which does not have constant latitude or longitude. This type of profile occurs in the line-of-sight, perspective, and radar terrain mask plots.

PTS obtains the elevation of each point along a profile by calling ALT (if a polynomial data base) or NALT (gridded data base). The elevations are stored in an array, IB. (See also the POINT Common Area).

(Drawing 6)

SUBROUTINE PTS(XL,AZ,DELTA,NPTS)

Parameters

XL - Two dimensional array containing the following values:

XL(1,1) - Latitude of initial profile point.

XL(2,1) - Longitude of initial profile point.

XL(1,2) - Latitude of final profile point.

XL(2,2) - Longitude of final profile point.

AZ - Bearing of profile in radians.

DELTA - Spacing between elevation points in meters.

NPTS - The number of elevation points along a profile.

PTSDR This subroutine is called by PTMN to obtain the elevations of points along a profile. Depending on the value of IFLAG, it calls either PTS or PTSSEC to get these elevations. The IFLAG value is assigned in PTMN depending on the type of plot being made.
(Drawing 6)

SUBROUTINE PTSDR(IFLAG,XL,AZ,DELTA,NPTS).
Parameters

IFLAG - Flag specifying subroutine to call
1 - Call PTS(XL,AZ,DELTA,NPTS)
2 - Call PTSSEC(XL,NPTS)

(See PTS and PTSSEC for descriptions of the variables XL,AZ,DELTA and NPTS).

PTSSEC This subroutine obtains elevations of points along a profile which has constant latitude or longitude. Profiles of constant latitude or longitude occur in the contour plot and the 3-dimensional plot. In these plot types the profiles are lines parallel to plot boundaries which are input by the user as constant latitude and longitude values.

PTSSEC obtains the elevation of each point along a profile by calling ALT (if a polynomial data base) or NALT (gridded data base). The elevations are stored in an array, IE. (See also the POINT Common Area).
(Drawing 6)

SUBROUTINE PTSSEC(XL,NPTS)
Parameters

XL - Two dimensional array containing the following values:

XL(1,1)	Latitude of initial profile point.
XL(2,1)	Longitude of initial profile point.
XL(1,2)	Latitude of final profile point.
XL(2,2)	Longitude of final profile point.

NPTS - The number of elevation points along a profile.

RDGPRT prints on the Tektronix screen the ridge line option previously chosen by the user in subroutine PLTOPT. The ridge line option is stored in the variable IRDGE in the RIDGE Common Area, as follows:

IRDGE = 0	All ridge lines will not be plotted.
IRDGE = 1	All ridge lines will be plotted.

PLTOPT automatically sets IRDGE = 1 for the range lines and range lines with grid plot options. For the other plot options (grid lines, contour levels, and contour levels with grid), the user specifies whether or not all ridge lines are to be plotted.
(Drawings 3,5)

SUBROUTINE RDGPRT
Parameters - none

RDHDFT This subroutine reads the header block (block 0) of the feature data file. It is called by DRIVFT which plots feature data on the contour plot or radar terrain mask plot. RDHDFT also computes a conversion flag, ICONV. (See also the PLOTVR Common Area). This variable specifies the type of conversion between WGS and UTM units to be made for the feature data in the file, as follows:

ICONV Value	Conversion
0	None
1	UTM to WGS
-1	WGS to UTM

(Drawing 12)

SUBROUTINE RDHDFT(IBUF)

Parameter

IBUF - 256 word array containing information from header of feature data file.

RMAIN is the main program for the radar terrain mask plot. RMAIN calls the subroutines which generate and display the RTM plot. The user may choose from the following options:

Safe area contours
Acquisition contours
Safe area below a given ceiling
Fields of fire

At the option of the user, corrections for the earth's curvature and/or atmospheric refractivity may be made. Feature data points can also be plotted on the radar terrain mask.

User inputs for the RTM plot are made in subroutine INRTM. (See also subroutines DRIVFT and ERTOPT.) (Drawing 4)

Parameters - none

RPARSM stores parameters needed to draw a radar terrain mask plot to scale.
(Drawing 8)

SUBROUTINE RPARSM(ZDIM,RLEN)
Parameters
ZDIM = the larger of the X or Y plot dimensions
RLEN = 11.0

RTMPLT assigns values to these variables before calling RPARSM. RPARSM calls PLOTB which stores ZDIM and RLEN in the IPBUF array; when this array is full, they are written to the plot file.

RPLOT does the plotting for the radar terrain mask plot as follows:

1. It calls RTMPLT which writes the title and relevant plot parameters on the plot.
2. It calls GRDRTM if a grid is to be drawn on the plot.
3. If feature data is to be plotted, RPLOT cmputes the values of the variables in the RTMPAR Common Area.
4. For each profile, it obtains the elevations of points along the profile by calling JPRFRD, which reads this information from the profile file.

   Based on this information, RPLOT draws the type of radar terrain mask plot specified by the user in INRTM, i.e., safe area contours, acquisition contours, etc.

(Drawing 8)

SUBROUTINE RPLOT
Parameters - none

RTM is a main program which calls the subroutines involved in the plotting for the radar terrain mask plot. It is swapped to from RMAIN. (See also subroutine RTSWP).

Before plotting is done, RTM reads necessary information from the disk file "SWPFL" and from the profile file. It then calls RPLOT to perform plotting and DRIVFT if feature data is to be plotted.
Finally, RTM calls PSWAP which swaps to the plotting programs in "PLTRN.SV".
(Drawings 4,8)

Parameters - none

RTMBD plots boundary type feature data on the radar terrain masking plot. This subroutine is called by PLBPFT.
(Drawing 12)

SUBROUTINE RTMBD(RY,RX,ICODE)
Parameters

RY - Latitude or northing of boundary point
RX - Longitude or easting of boundary point
ICODE - Boundary type code. (See SPSYM for a description of ICODE).

RTMFT plots single point feature data on the radar terrain masking plot. Each point inside the plot boundaries is plotted along with its appropriate military symbol. (See SPSYM). (Drawing 12)

SUBROUTINE RTMFT(RY,RX,ICODE)

Parameters

RY	-	Latitude or northing of boundary point
RX	-	Longitude or easting of boundary point
ICODE	-	Boundary type code

RTMPLT This subroutine does the initial plotting for the radar terrain masking plot type. The title, observer data, data on the area to be covered, and the contour levels are written around the plotting area. The observer's coordinates are determined so that the plot is centered in an 8 x 8 inch area in the middle of the Tektronix screen. (Drawing 8)

SUBROUTINE RTMPLT(AZ,DAZ,SC)

Parameters

AZ	-	Bearing in radians of the first radial
DAZ	-	The spacing in radians between radials
SC	-	The plot scale to be used for all plots (inches/meter)

RTSWP For the radar terrain masking plot, RMAIN, uses subroutine RTSWP for swapping to "RTM.SV" to do the radar terrain mask plotting.

Before swapping, RTSWP fills the ICOM array (See SWPCM Common Area) with variables from relevant common blocks and writes the array to block 0 of the disk file "SWPFL". This information is subsequently read from the disk by RTM and used in drawing the plot.
(Drawing 4)

SUBROUTINE RTSWP
Parameters - none

SCALFT Although it is still in the code, this subroutine is obsolete and should be removed. It is called by DRIVFT and was originally intended to compute boundaries for the 3-dimensional plot. This is now performed by SCALTH in program FTMAIN. DRIVFT is now used only for contour and radar terrain mask plots.
(Drawing 12)

SUBROUTINE SCALFT
Parameters - none

SCALPR This subroutine calculates the parameters needed to plot grid lines and feature data on perspective plots. (See the PARPRS Common Area for a description of these parameters).
(Drawing 11)

SUBROUTINE SCALPR
Parameters - none

SCALTH calculates the parameters needed to plot grid lines on three dimensional view plots. These parameters depend on the view face chosen by the user in INTHRD. (See also the CONTHR and THRBLK Common Areas).
(Drawing 11)

SUBROUTINE SCALTH
Parameters - none

SECDMS This subroutine converts a latitude or longitude in signed seconds into degrees, minutes, seconds and direction.
(Drawings 1,2,3,4,5,13)

SUBROUTINE SECDMS(K,XL,LDMS)
Parameters

K - Latitude or longitude indicator
1 - Latitude
2 - Longitude

XL - Latitude or longitude in signed seconds

LDMS(4) - Latitude or longitude in degrees, minutes, seconds, and direction.

SETUP This subroutine asks the user to input the name of the data base file and also the feature data file name if feature data is to be plotted. The main plotting programs, CMAIN,LMAIN,PMAIN,RMAIN and TMAIN call SETUP before additional user inputs are made in other subroutines.

SETUP also requests the user to specify whether input data will be in the WGS or mil grid system. (See the UNITS Common Area). If data is to be entered in mil grid units, SETUP calls MGSET.

(Drawings 1,2,3,4,5)

SUBROUTINE SETUP

Parameters - none

SODDIR Given the geographic location of a reference point, azimuth angle, and a distance along the azimuth to a second point, SODDIR computes the geographic position of the second point.

(Drawing 6)

SUBROUTINE SODDIR(XL,ALFAX,DISTX,PHI2,LAM2,IC)

Parameters

XL - XL(1) Reference point latitude (radians)
XL(2) Reference point longitude (radians)

ALPHAX - The azimuth angle (radians)

DISTX - Distance from the reference point (meters)

IC - Specifies the type of units
1 = WGS
2 = Mil Grid

PHI2 - Latitude of new point (radians)

LAM2 - Longitude of new point (radians)

SODINV uses parameters provided by INVM to calculate the azimuth angle and geodetic distance between two geographic points. This calculation is used by the line-of-sight plot program, LMAIN.
(Drawing 2)

SUBROUTINE SODINV(A,BO,XLAT1,XLONG1,XLAT2, XLONG2,GEODIS,AZ12,AZ21)
Parameters (All are double precision variables.)

A	=	63781350D0
BO	=	6356750.52D0
XLAT1	=	Latitude of position 1
XLONG1	=	Longitude of position 1
XLAT2	=	Latitude of position 2
XLONG2	=	Longtidue of position 2
GEODIS	=	The distance in meters
AZ12	=	Forward azimuth
AZ21	=	Back azimuth

SPSYM This subroutine is used to plot military symbols for single point feature data. These symbols can be drawn on all plot types except line-of-sight.
(Drawings 11,12)

SUBROUTINE SPSYM(X,Y,HT,ICODE,IDES)
Parameters

X and Y are the coordinates about which the symbol is centered.

HT	-	Height of the symbol.
ICODE	-	Symbol Code (See below).
IDES	-	Description of the size of the military unit. (See the DESCRI Common Area).

The Unit symbol is a box, which is filled in as follows according to the value of ICODE:

ICODE Value	Military Unit
2	Infantry
3	Field Artillery
4	Engineer
5	Armor
19	Urban Area

After plotting the appropriate military unit symbol, SPSYM calls SZPLOT which plots the unit size symbol above the unit symbol.

SRCHFI searches a feature data file for a particular record and computes the location of the record within the file. It is used by the programs in "MAINFT.SV" for creating and modifying feature data files.

(Drawing 13)

SUBROUTINE SRCHFI(IBLK,LSTBLK,IREC,IWD)

Parameters

LSTBLK	-	number of the last block in the feature data file.
IREC	-	record in the file to search for.
IBLK	-	Block number where record begins.
IWD	-	IBLK where record begins.

STAT prints out the statistics on the use of the data base by the programs which compute the terrain profiles. This information is stored in the PNTS Common Area and is as follows:

1. The number of points computed from the disk (i.e., data base) file.
2. The number of po⁊nts requested from outside the data base.

(Drawings 1,2,3,4,5)

SUBROUTINE STAT

Parameters - none

SZPLOT This subroutine plots the Unit size symbol above the military Unit symbol already plotted by SPSYM OR UNPLT. (Drawings 11,12)

SUBROUTINE SZPLOT(XPT,YPT,IDES,WIDTH,THETA)

Parameters

XPT and YPT are the coordinates about which the size symbol is centered.

IDES - Description of the size of a military Unit. (See the DESCRI Common Area).

WIDTH - The width of the Unit Symbol.

THETA - The angle at which the size symbol will be drawn on the plot normally 0.0 degrees).

THKPLT This subroutine plots a line of specified thickness between two points. It is currently used in drawing the radar terrain mask plot.
(Drawing 8)

SUBROUTINE THKPLT(X1,Y1,X2,Y2,TH)

Parameters

X1 - X coordinate of first point
Y1 - Y coordinate of first point
X2 - X coordinate of second point
Y2 - Y coordinate of second point
TH - Desired thickness of line in inches.

THMBPT This subroutine is used in the process of digitizing feature data. Feature data can be digitized (not currently implemented) or input by typing it in from the terminal. (See the programs in "MAINFT.SV" for creating and modifying feature data files). THMBPT turns on the thumb wheel cursor and reads the cursor position when any key is pressed.
(Drawing 13)

SUBROUTINE THMBPT(ICHAR,IX,IY)

Parameters

IX - X coordinate of cursor
IY - Y coordinate of cursor
ICHAR - Code for the key that was pressed.

THRBD plots boundary type feature data on the 3-dimensional plot type. This subroutine is called by DRFTTH.
(Drawing 11)

SUBROUTINE THRBD(RY,RX,ICODE)

Parameters

RY	-	Latitude or northing of boundary point
RX	-	Longitude or easting of boundary point
ICODE	-	Boundary type code. (See SPSYM for a description of ICODE).

THREED is a main program which calls the subroutines involved in the plotting for the 3-dimensional plot. It is swapped to from TMAIN. (See also THSWP).
Before plotting is done, THREED reads necessary information from the disk file "SWPFL" and from the profile file. It then calls TPLOT to perform plotting and calls FTSWP if feature data is to be plotted.

Finally, THREED calls PSWAP which swaps to the plotting programs in "PLTRN.SV".
(Drawings 5,9)

Parameters - none

THRFT plots single point feature data on a 3-dimensional plot. Each point not hidden by terrain features is plotted along with its appropriate military symbol. (See SPSYM).
(Drawing 11)

SUBROUTINE THRFT(RY,RX,ICODE)

Parameters

RY - Latitude or northing of feature point.

RX - Longitude or easting of feature point.

ICODE - Code for single point feature. (See SPSYM for a description of ICODE).

THRPLT This subroutine does initial plotting for the 3-dimensional plot. It also computes the plot scale and the angle the projection plane makes with the horizontal. THRPLT is called by subroutine TPLOT.
(Drawing 9)

SUBROUTINE THRPLT(XLTH,YLTH,SC,THETA)

Parameters

XLTH - Length of boundary parallel to projection plane (meters).

YLTH - Length of boundary not parallel to projection plane (meters).

SC - Plot scale in inches per meter.

THETA - Tilt of projection plane from horizontal (radians).

THSWP TMAIN uses THSWP for swapping to "THREED.SV" where the the 3-dimensional view plotting is done.

Before swapping, THSWP fills the ICOM array (See SWPCM Common Area) with variables from relevant Common blocks and writes the array to block 0 of the disk file "SWPFL". This information is subsequently read from the disk by THREED and used in drawing the plot.
(Drawing 5)

SUBROUTINE THSWP

Parameters - none

TITLE This subroutine prints a title page on the Tektronix terminal at the beginning of each run. It is used by all five plot types.

The title page displays information about the data base file being used for the plot.

(Drawings 1,2,3,4,5)

SUBROUTINE TITLE

Parameters - none

TMAIN is the main program for the 3-dimensional (oblique view) plot. An oblique view is a projection of an area of terrain onto a plane with the profiles parallel to each other but not necessarily perpendicular to the plane. The plane is parallel to the closest boundary of the terrain. TMAIN calls subroutines which generate the 3-dimensional view. The user may choose from the following plot options:

Grid lines
Range lines
Contour levels
Range lines with grid lines
Contour levels with grid lines

At the option of the user, corrections for the earth's curvature and/or atmospheric refractivity may be made. Feature data points can also be plotted on the 3-dimensional view.

User inputs for the 3-dimensional plot are made in subroutine INTHRD. (See also subroutines ERTOPT,FTMAIN, and PLTOPT.) (Drawing 5)

Parameters - none

TPARSM stores parameters in the plot file needed to draw a 3-dimensional plot to scale.

(Drawing 9)

SUBROUTINE TPARSM(XLEN,SC)

Parameters

XLEN	-	Width of plot (inches)
SC	-	Plot scale (inches/meter)

THRPLT assigns values to these variables before calling TPARSM. TPARSM calls PLOTB which stores XLEN and SC in the IPBUF array; when this array is full, they are written to the plot file.

TPLOT does the plotting for the 3-dimensional plot as follows:

1. It calls THRPLT which does initial plotting and computes the plot scale.
2. For each profile, it obtains the elevations of points along the profile by calling PRFRD, which reads the information from the profile file. Based on this information, TPLOT computes coordinates of each point along the profile. Points which are hidden by terrain features are detected so that these will not be drawn on the plot. (See also the HIDCM Common Area).
3. If feature data or a grid is to be drawn on the plot, TPLOT writes the coordinates of the points to the Points File for later use by the subroutines which plot the grid or feature data.

(Drawing 9)

SUBROUTINE TPLOT

Parameters - none

UNPLT This subroutine plots the military Unit size symbol between two boundary points (boundary type feature data). See also SZPLOT and the DESCRI Common Area.
(Drawings 11,12)

SUBROUTINE UNPLT(PO,PN)

Parameters

PO - X,Y coordinate of first boundary point.

PN - X,Y coordinate of second boundary point.

Note: UNPLT is not currently being loaded into any of the executable programs. It is to be implemented as part of the feature data plotting programs.

UTMPRT converts a floating point number into an alphanumeric array of length NCHAR, with an Al format. This is done to preserve the precision in the printing of large floating point numbers.
(Drawings 1,2,3,4,5)

SUBROUTINE UTMPRT(UTM,IU,NCHAR)

Parameters

UTM - Floating point number

NCHAR - Number of desired characters or places in number.

IU - Alphanumeric character array.

UTM2LL This subroutine computes the geographic coordinates (latitude and longitude) of a point whose Universal Transverse Mercator (UTM) coordinates are known. UTM2LL is a utility type subroutine used by all five plot types. (See also the PARAM Common Area).

(Drawings 1,2,3,4,5,6,11,12)

SUBROUTINE UTM2LL(L,IH,XNOR,XEAST,IZONE,XLTS,XLNGS)

Parameters

L - Specifies the spheroid to be used in the transformation of coordinates.

IH - 1 = Northern Hemisphere
2 = Southern Hemisphere

XNOR - Is the northing value for the point.

XEAST - Is the easting value for the point.

IZONE - Gives the UTM zone number.

XLTS - Is the latitude of the point in signed seconds.

XLNGS - Is the longitude of the point in signed seconds.

UTM2MG This subroutine converts a UTM value to a mil grid value that is in character format. It is used by programs in "MAINFT.SV" which create and modify feature data files.
(Drawing 13)

SUBROUTINE UTM2MG(UTMN,UTME)

Parameters

UTMN - UTM Northing

UTME - UTM Easting

Note: The output from this subroutine is passed through Common Area IMGCOM. (See the IMGCOM Common Area).

VCON This subroutine plots the contour levels between two profiles on perspective and 3-D plots (See subroutine CON for contour and radar terrain mask plots.)
(Drawings 7,9)

SUBROUTINE CON(IX,IY,IZ,NPTS)

Parameters

IX,IY are 256 x 2 arrays of X,Y coordinates

IZ is 256 x 2 array containg the 2 coordinates for a profile. This array is obtained from subroutine PRFRD (for 3-D plots) or from subroutine JPRFRD (for perspective plots).

NPTS is the number of points along the profile.

VPL2 is the main program which calls the routines which display a plot file on the Versatec plotter. It is swapped to from "PLTRN.SV". (See V2SWP).

Before the plot is displayed, VPL2 reads block 0 of the disk file "V2SFL" and initializes variables based on this information. It then opens either the permanent plot file (if PLTRN is being run as a stand alone program) or the temporary plot file "PLOTF" (if PLTRN was swapped to from the plotting programs).

VPL2 then reads as necessary from the plot file and calls the appropriate Versatec plotting routines.

(Drawings 10,14)

Parameters - none

VPRMS This subroutine sets up certain parameters needed by the Versatec plotter. It computes Versatec scale factors, window limits, etc. PLTRN calls VPRMS to obtain these parameters before swapping to the Versatec plotting programs in "VPL2.SV". (See also the VPRM Common Area).

(Drawing 10)

SUBROUTINE VPRMS

Parameters - none

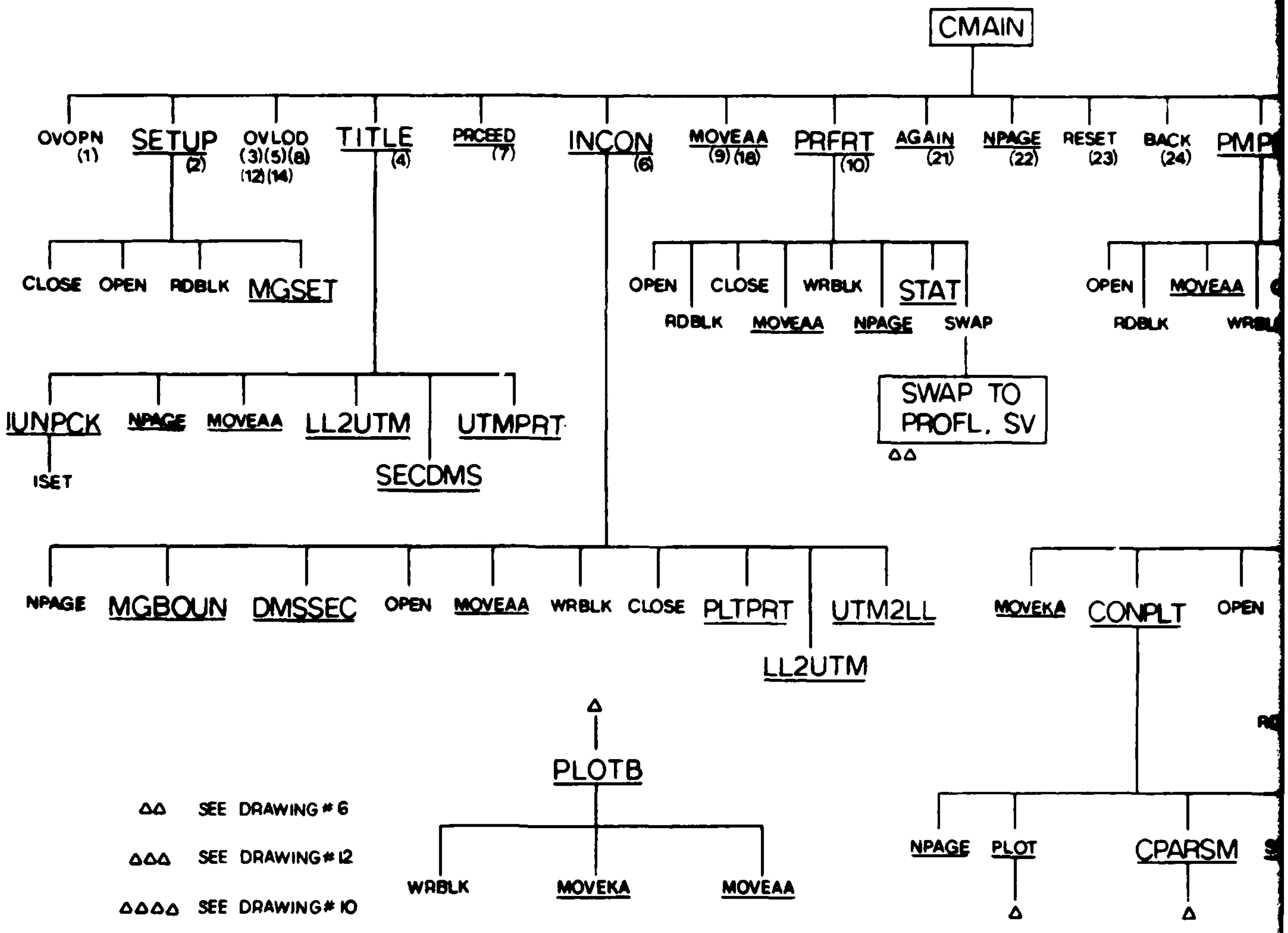
V2SWP This subroutine is used to swap to "VPL2.SV" which contains the routines for displaying a plot file on the Versatec plotter.
Before swapping, V2SWP fills the ICOM array with variables from the Common blocks and writes this information to block 0 of the disk file "V2SFL", where it is subsequently read and used by VPL2.
(Drawing 10)

SUBROUTINE V2SWP
Parameters - none

WINDOW This subroutine puts a user-defined area of the data base into core, so that the program will run faster when accessing that area by minimizing disk access. Currently this process is used only for polynomial data bases.
(Drawing 6)

SUBROUTINE WINDOW(IFLAG,XL)
Parameters
IFLAG - 0 = Contour plot or 3-D plot-boundaries are of constant latitude and longitude.
1 = RTM plot or perspective plot-boundaries don't have constant latitude and longitude, so program must calculate average positions.
XL - 4-element array holds the boundary coordinates of window.

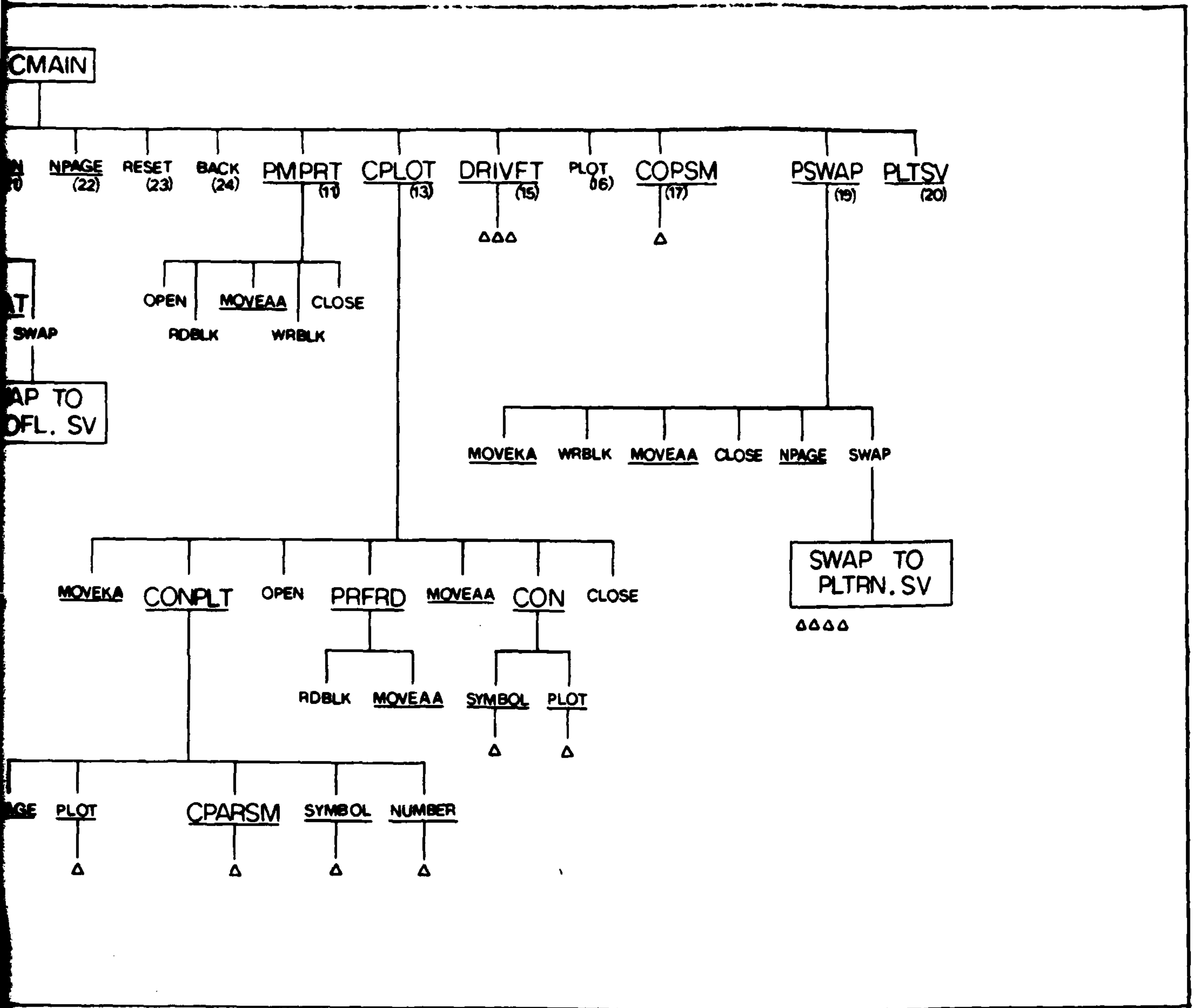
IID. CALLING SEQUENCE FLOW DIAGRAMS



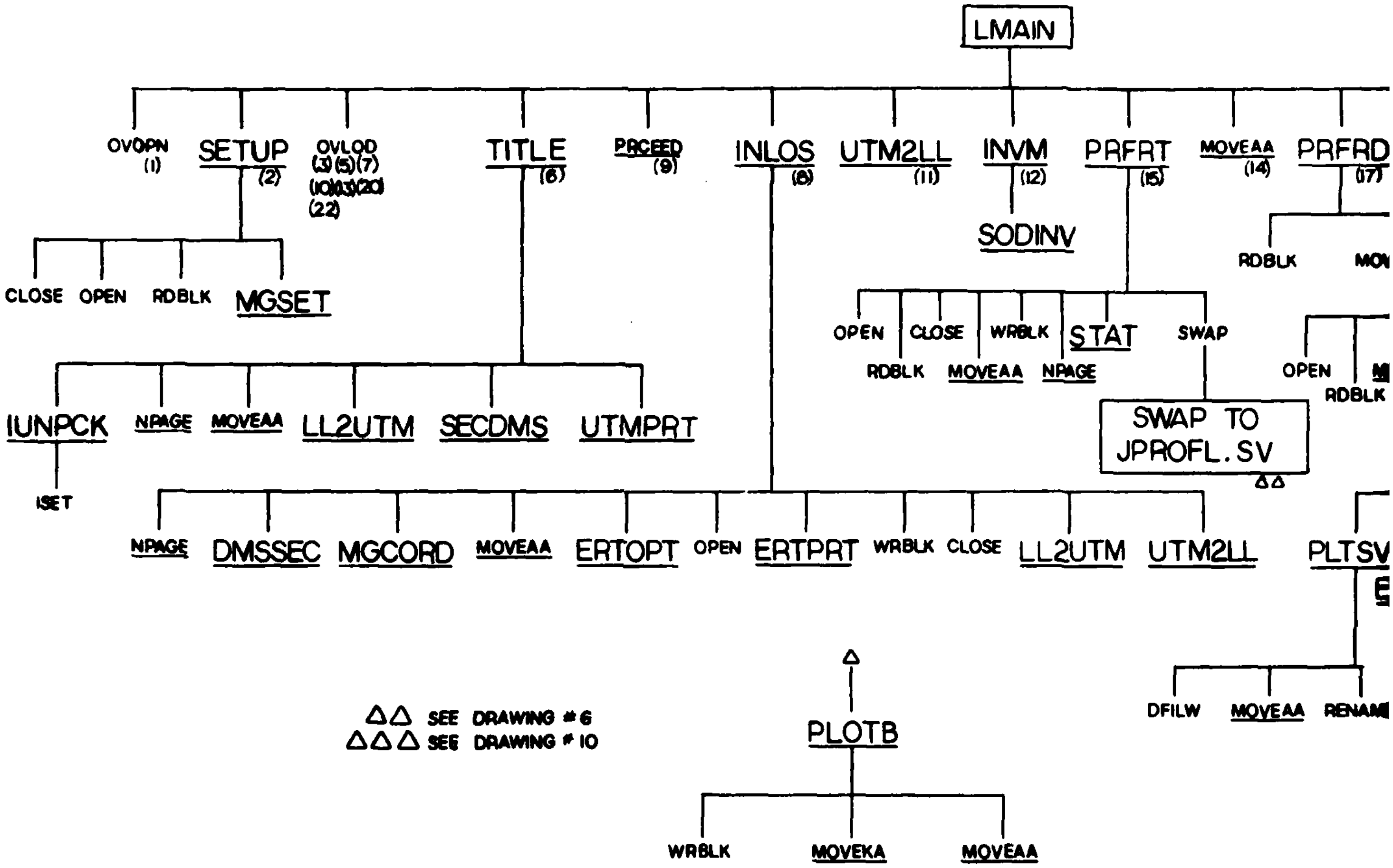
CLOAD.SV

SUBROUTINE #	FUNCTION
1-4	INITIALIZE
5-7	READ IN CONTOUR INPU
8-11	CALCULATE PROFILE E
12-13	RETRIEVE PROFILES AND
14-15	DO FEATURE DATA PLOT
16-17	MAKE COPIES OF PLOT
18-19	COMPLETE PLOT FILE A
20	OPTION TO SAVE THE
21-24	CONTINUE FOR ANOTHER ELSE STOP PROGRAM

1



NE #	FUNCTION	OVERLAY USED (CLOAD,OL)
	INITIALIZE	← TITOL
	READ IN CONTOUR INPUTS	← INCOL
	CALCULATE PROFILE ELEVATION	← PRFOL
	RETRIEVE PROFILES AND DO PLOTTING	← CPTOL
	DO FEATURE DATA PLOTTING	← FETOL
	MAKE COPIES OF PLOT	
	COMPLETE PLOT FILE AND SWAP TO WPLTRN.SV	
	OPTION TO SAVE THE PLOT FILE	
	CONTINUE FOR ANOTHER CONTOUR PLOT, ELSE STOP PROGRAM	

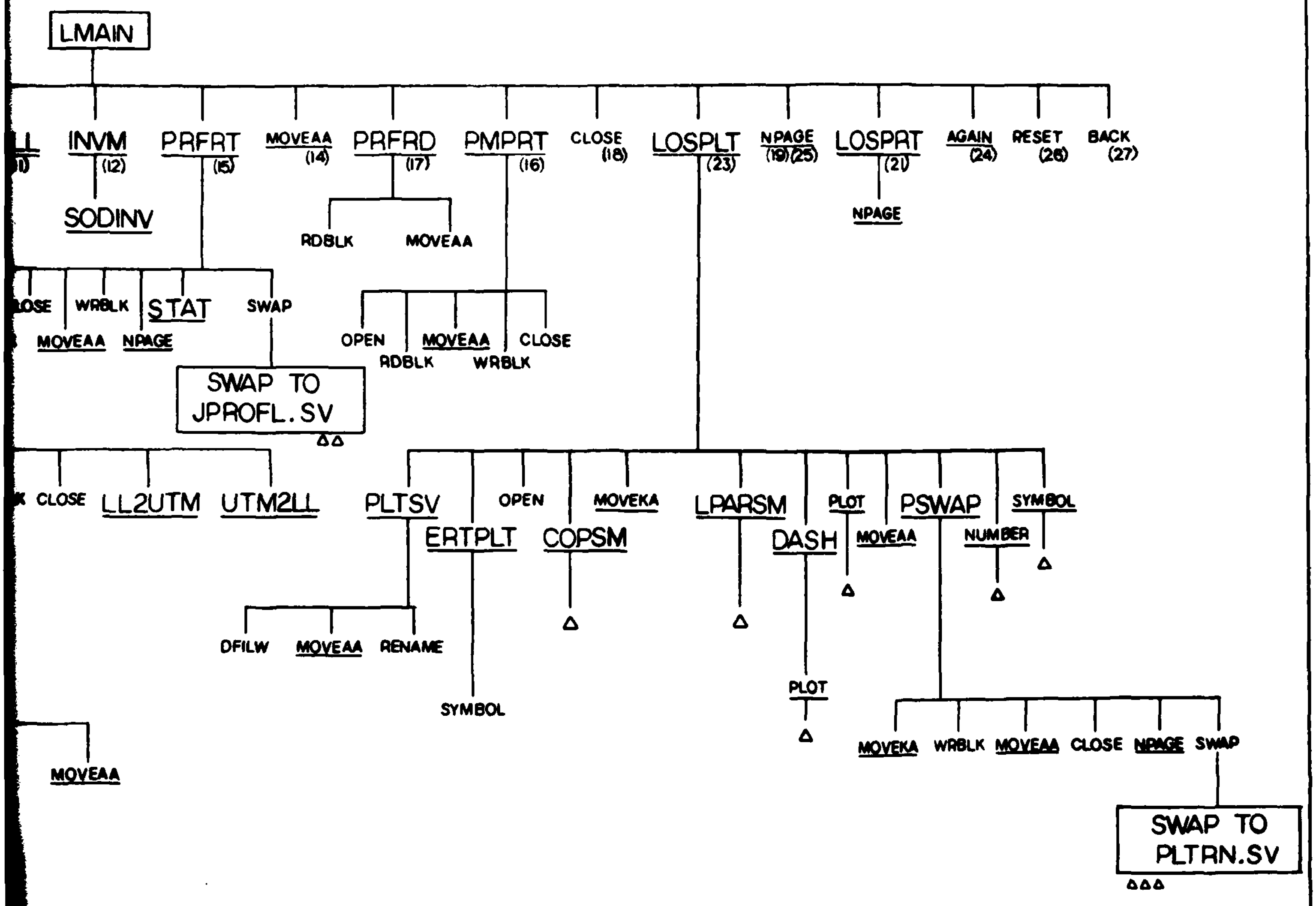


ΔΔ SEE DRAWING #6
ΔΔΔ SEE DRAWING # 10

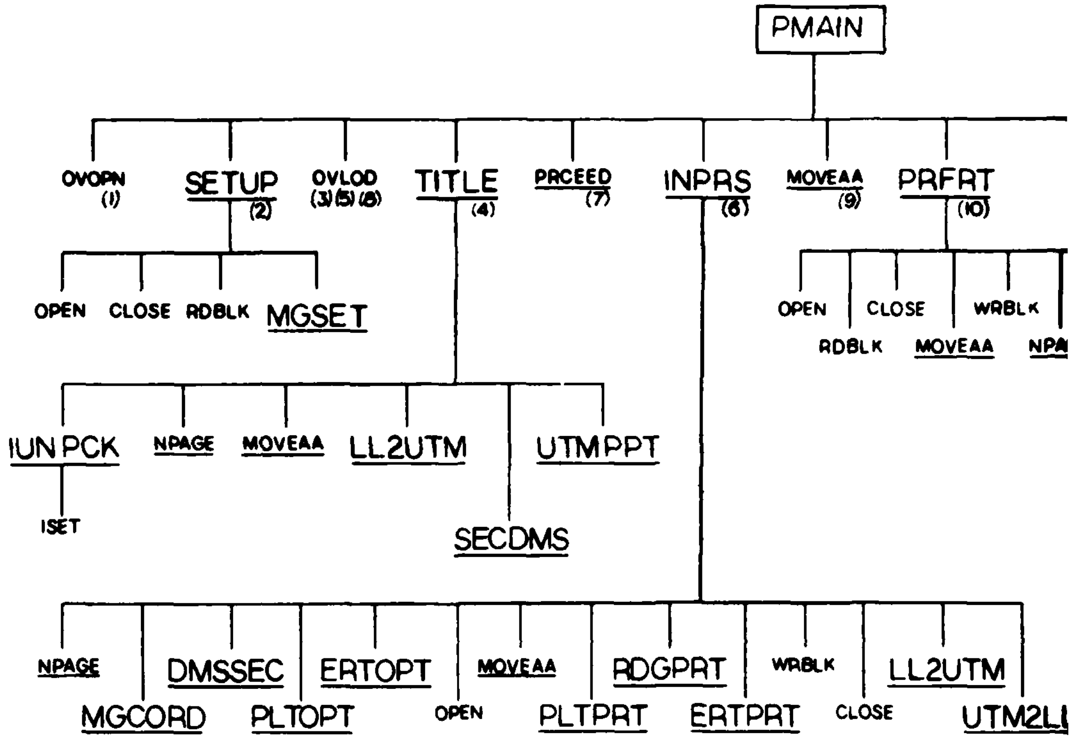
LLOAD.SV

SUBROUTINE #	FUNCTION
1-6	INITIALIZE AND WRITE TITLE
7-9	READ IN L.O.S. INPUTS
10-12	CALCULATE DISTANCE AND
13-16	CALCULATE PROFILE ELEVATI
17-18	RETRIEVE PROFILE
19-21	PRINT TABLE OF ELEVATIONS
22-23	PLOT LINE OF SIGHT PROFIL
24-27	CONTINUE FOR ANOTHER L PLOT, ELSE STOP PROGRAM

2



SUBROUTINE #	FUNCTION	OVERLAY USED (LLOAD.OL)
1-6	INITIALIZE AND WRITE TITLE	← PLTOL, TITOL
7-9	READ IN L.O.S. INPUTS	← ILSOL
10-12	CALCULATE DISTANCE AND BEARING	← INVOL
13-16	CALCULATE PROFILE ELEVATIONS	← PRFOL
17-18	RETRIEVE PROFILE	"
19-21	PRINT TABLE OF ELEVATIONS	← LPROL
22-23	PLOT LINE OF SIGHT PROFILE	← LPLOL
24-27	CONTINUE FOR ANOTHER L.O.S. PLOT, ELSE STOP PROGRAM	



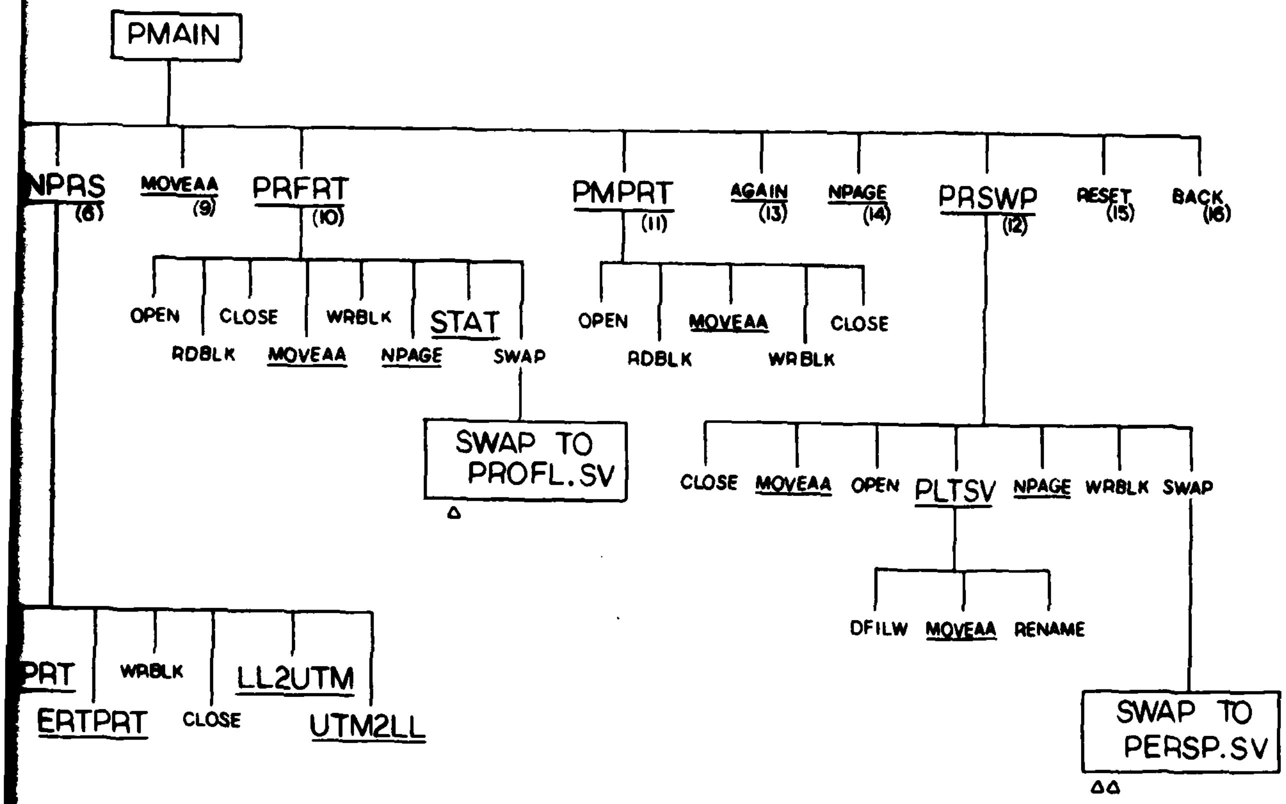
△ SEE DRAWING # 6
△△ SEE DRAWING # 7

PLOAD.SV

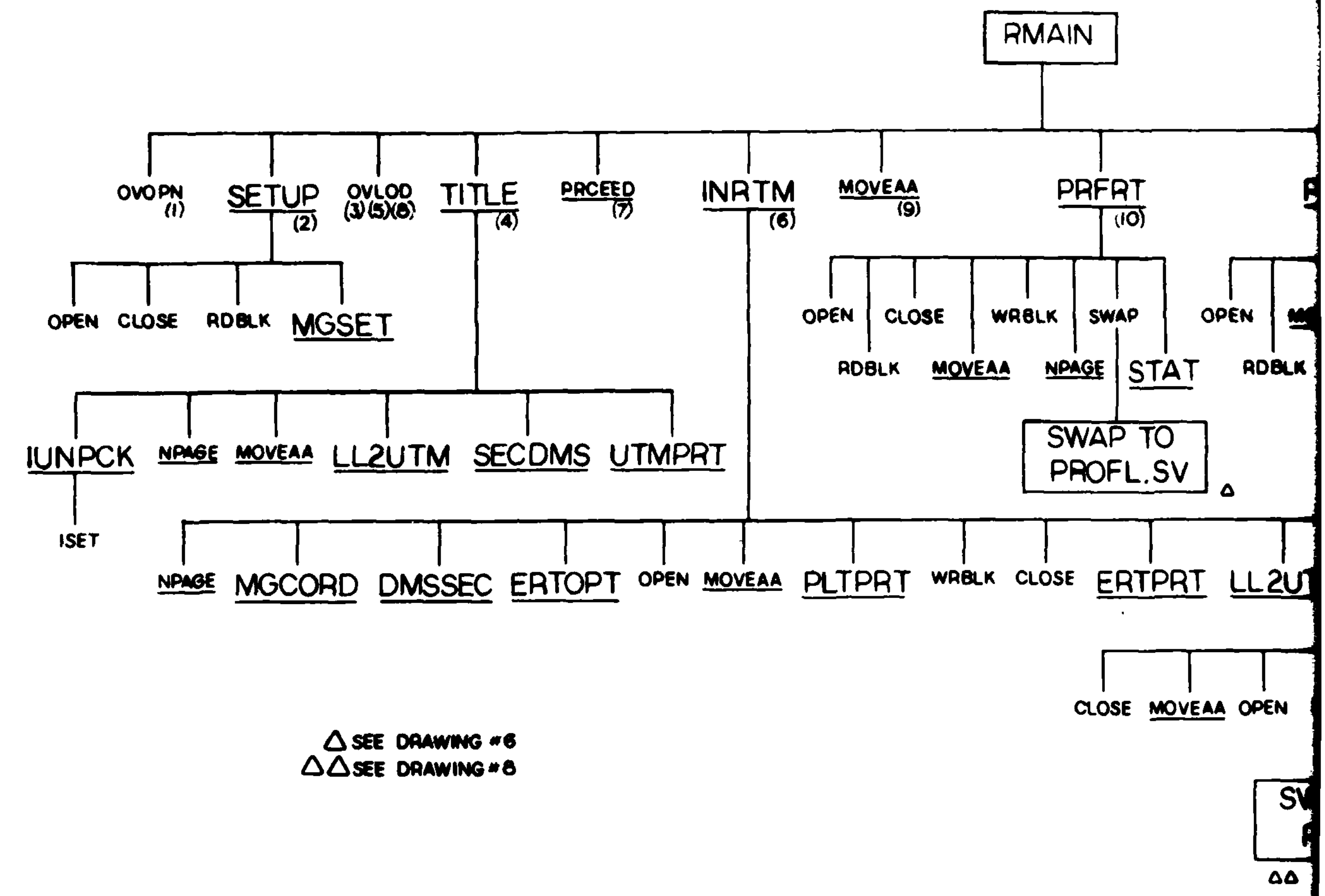
SUBROUTINE #	FUNCTION
1-4	INITIALIZE AND WRITE TIT
5-7	READ IN PERSPECTIVE VI
8-11	CALCULATE PROFILE ELEV
12	RETRIEVE PROFILES AND
13-16	CONTINUE FOR ANOTHER PLOT,ELSE STOP PRO

3

1



OUTINE #	FUNCTION	OVERLAY USED (PLOAD.OL)
1-4	INITIALIZE AND WRITE TITLE	← TITOL
5-7	READ IN PERSPECTIVE VIEW INPUTS	← IPROL
8-11	CALCULATE PROFILE ELEVATIONS	← PRFOL
12	RETRIEVE PROFILES AND DO PLOTTING	
13-16	CONTINUE FOR ANOTHER PERSPECTIVE PLOT, ELSE STOP PROGRAM	

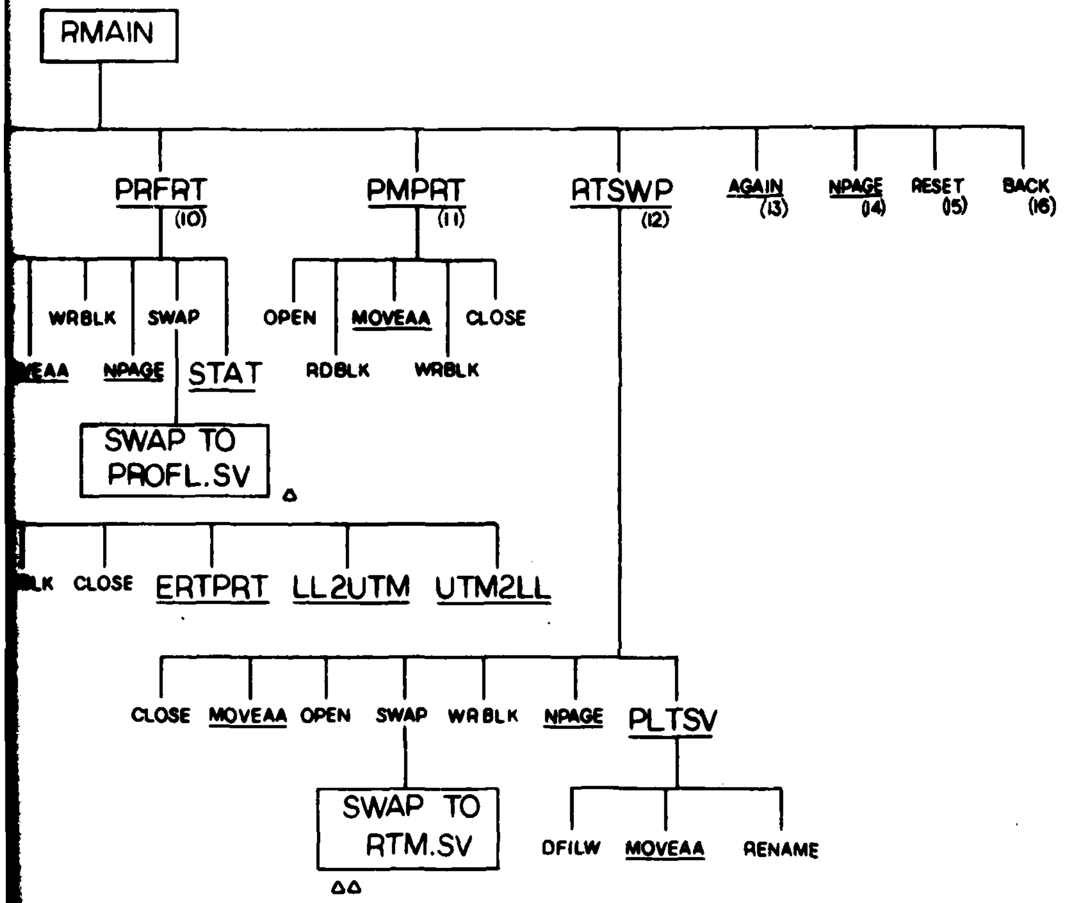


△ SEE DRAWING #6
△△ SEE DRAWING #8

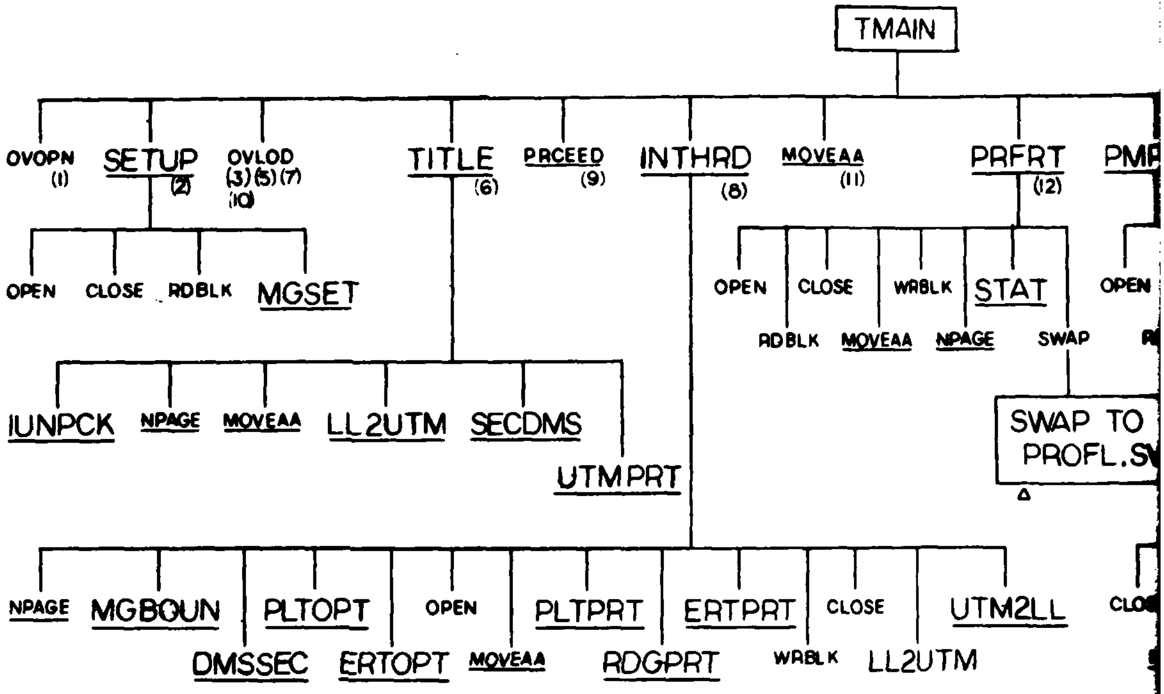
SUBROUTINE#	FUNCTION
1-4	INITIALIZE AND WRITE T
5-7	READ IN RADAR TERRA
8-11	CALCULATE PROFILE E
12	RETRIEVE PROFILES AN
13-16	CONTINUE FOR ANOTH MASK PLOT, ELSE S

RLOAD.SV

4



NE#	FUNCTION	OVERLAY USED (RLOAD.OL)
	INITIALIZE AND WRITE TITLE	← TITOL
	READ IN RADAR TERRAIN INPUTS	← IRTOL
	CALCULATE PROFILE ELEVATIONS	← PRFOL
	RETRIEVE PROFILES AND DO PLOTTING	
	CONTINUE FOR ANOTHER RADAR TERRAIN MASK PLOT, ELSE STOP PROGRAM	

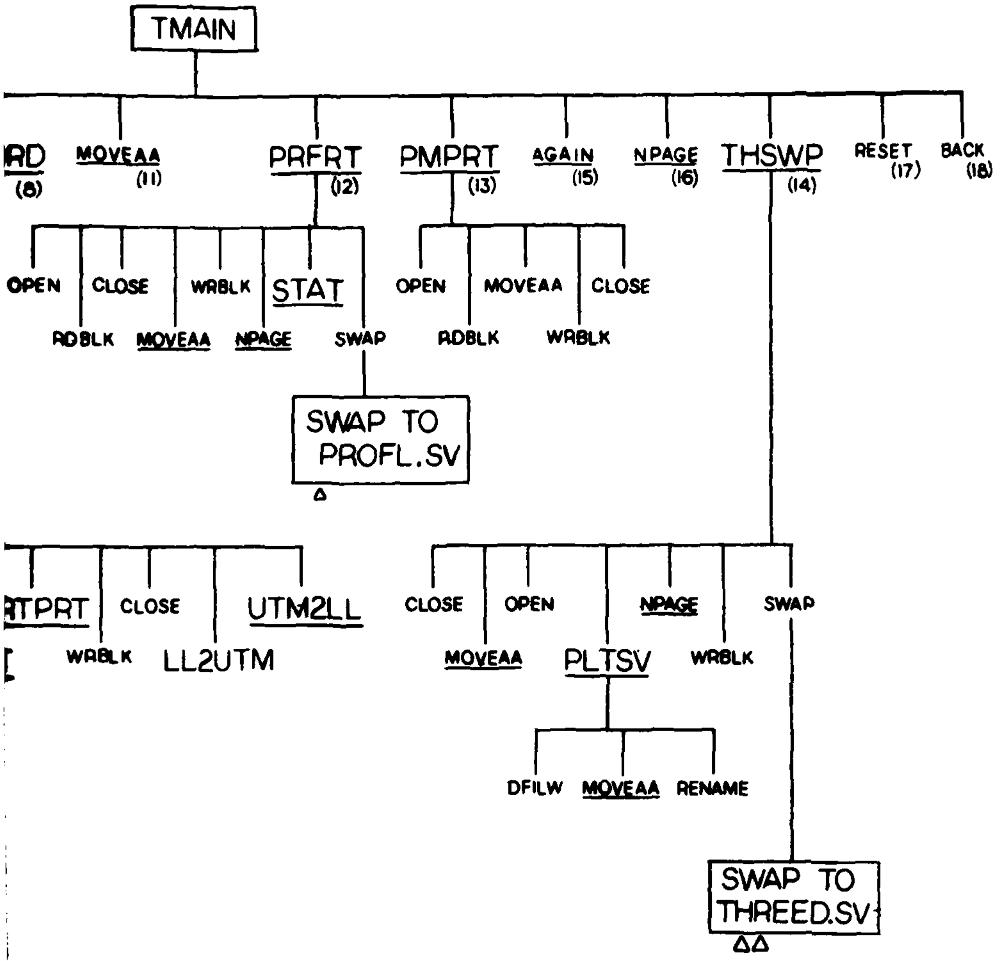


Δ SEE DRAWING #6
ΔΔ SEE DRAWING #9

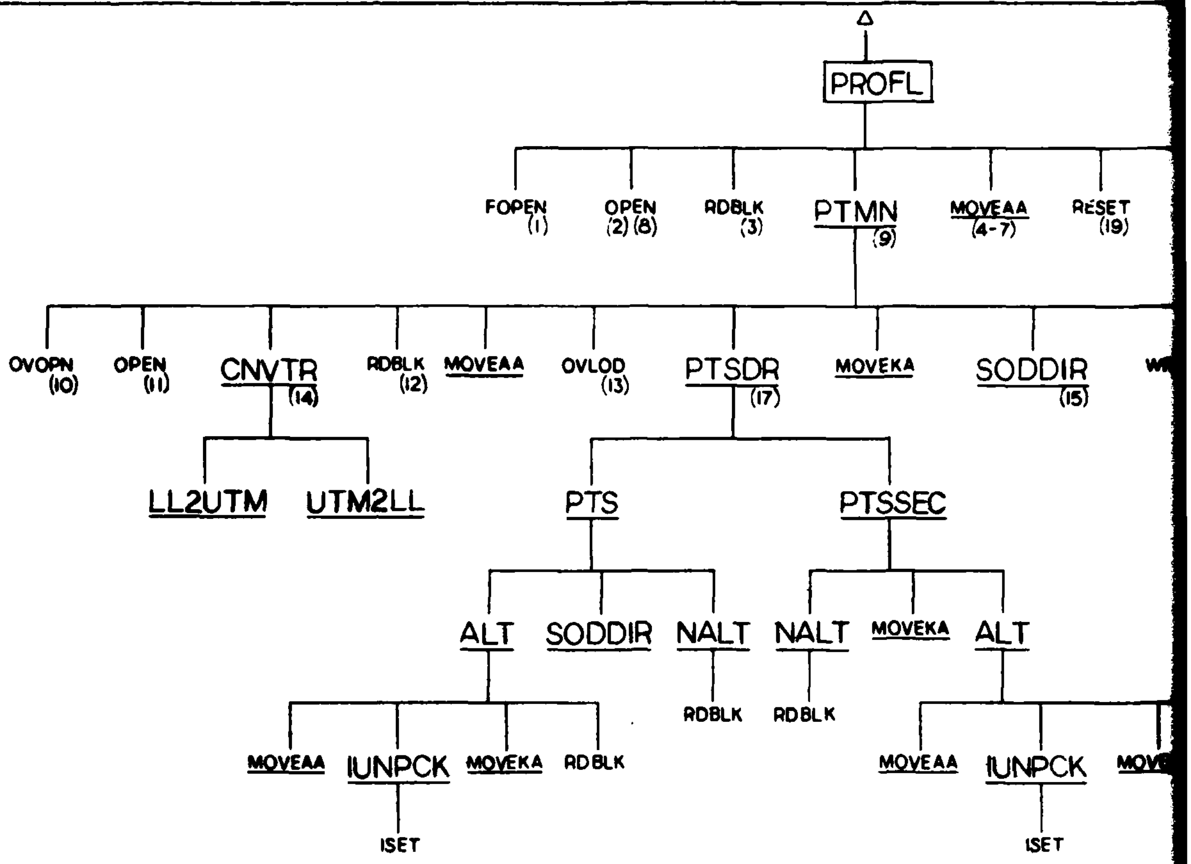
SUBROUTINE #	FUNCTION
1-6	INITIALIZE AND WRITE
7-9	READ IN 3-DIMENSION
10-13	CALCULATE PROFILE
14	RETRIEVE PROFILES A
15-18	CONTINUE FOR ANOT PLOT, ELSE STOP P

TLOAD.SV

5



SUBROUTINE #	FUNCTION	OVERLAY USED (TLOAD.OL)
1-6	INITIALIZE AND WRITE TITLE	← PLTOL, TITOL
7-9	READ IN 3-DIMENSIONAL VIEW INPUTS	← ITHOL
10-13	CALCULATE PROFILE ELEVATIONS	← PRFOL
14	RETRIEVE PROFILES AND DO PLOTTING	
15-18	CONTINUE FOR ANOTHER 3 DIMENSIONAL PLOT, ELSE STOP PROGRAM	

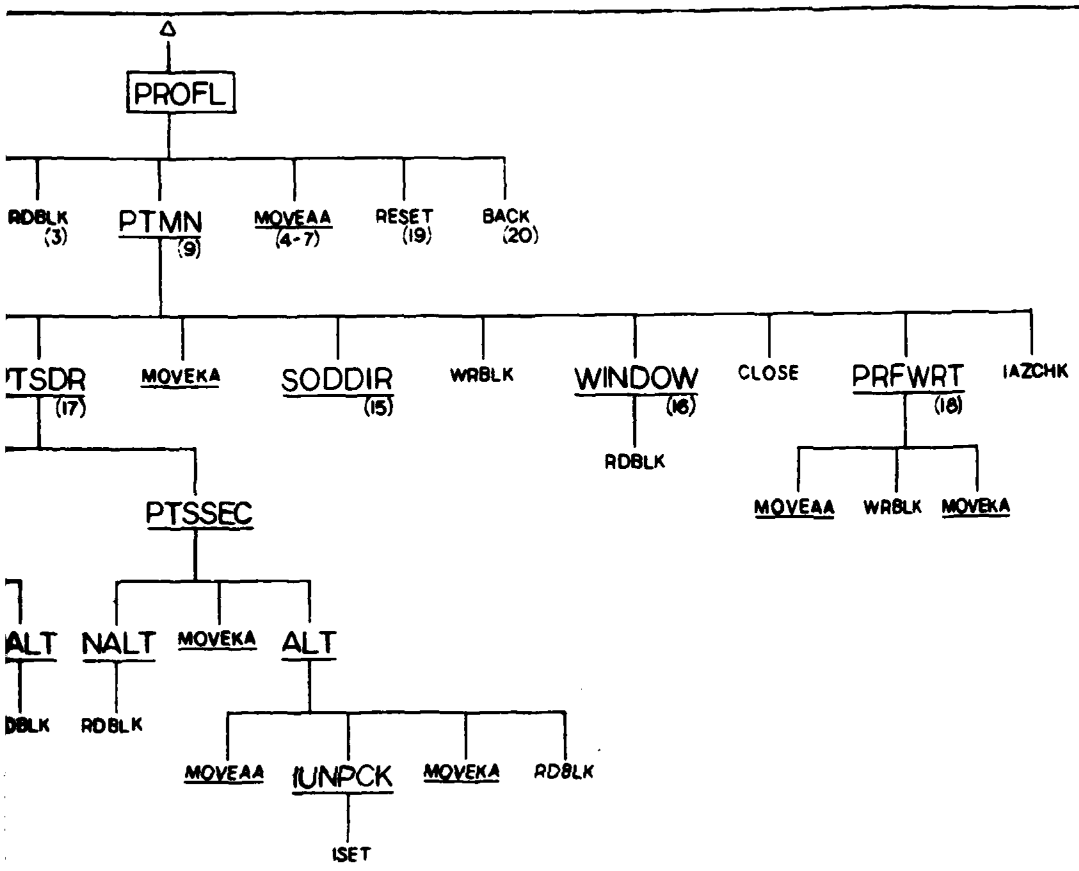


△ FROM DRAWINGS # 1,2,3,4,5

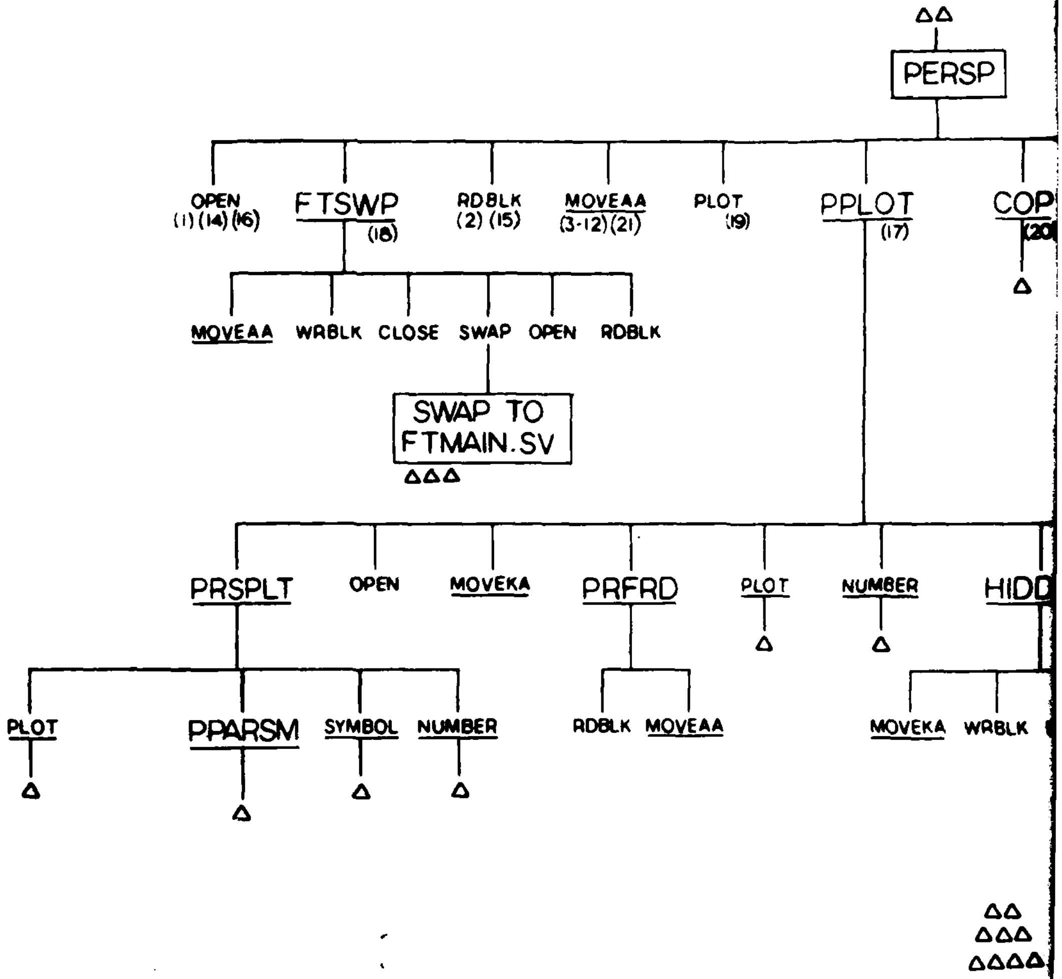
SUBROUTINE#	FUNCTION
1-7	OPEN SWAP FILE AND READ HEA
8	OPEN PROFILE FILE
9	EXTRACT PROFILES
10-12	READ HEADER OF DATA BASE FIL
	FOR EACH PLOT TYPE
13	LOAD APPROPRIATE OV
14	COORDINATE CONVERSI
15	COORDINATE TRANSFOR
16	USE CORE FOR GREATER
17	OBTAIN ELEVATIONS O
18	WRITE PROFILE ELEVAT
19-20	RETURN TO CALLING PROGRAM

PROFL.SV

6



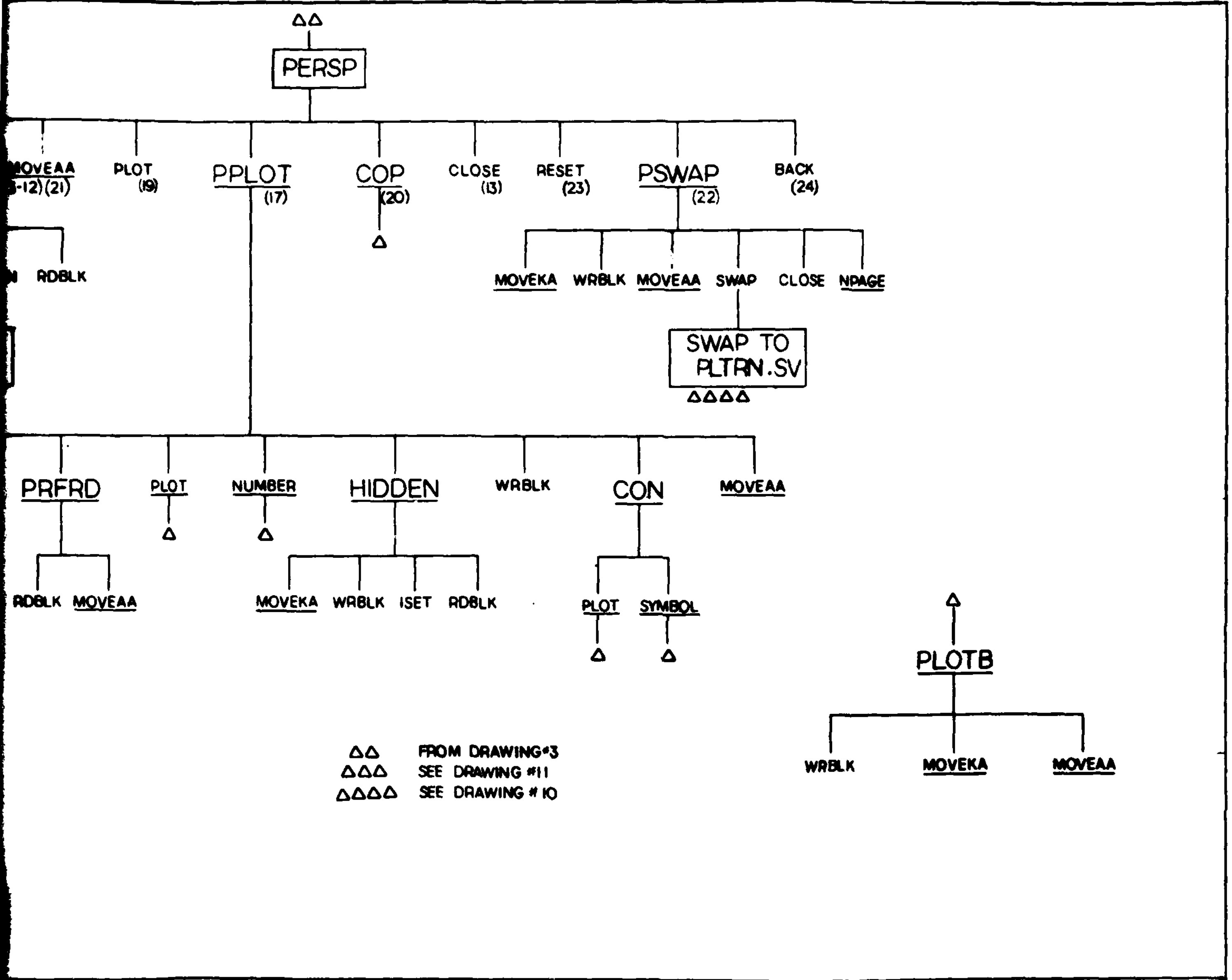
ROUTINE#	FUNCTION	OVERLAY USED
1-7	OPEN SWAP FILE AND READ HEADER	
8	OPEN PROFILE FILE	
9	EXTRACT PROFILES	
10-12	READ HEADER OF DATA BASE FILE	(JPROFL.OL)
	FOR EACH PLOT TYPE	
13	LOAD APPROPRIATE OVERLAYS	
14	COORDINATE CONVERSIONS	← CVTOL
15	COORDINATE TRANSFORMATIONS	
16	USE CORE FOR GREATER SPEED	← WNDOL
17	OBTAIN ELEVATIONS OF POINTS ALONG PROFILES	← PTSOL
18	WRITE PROFILE ELEVATIONS ONTO DISK	
19-20	RETURN TO CALLING PROGRAM	



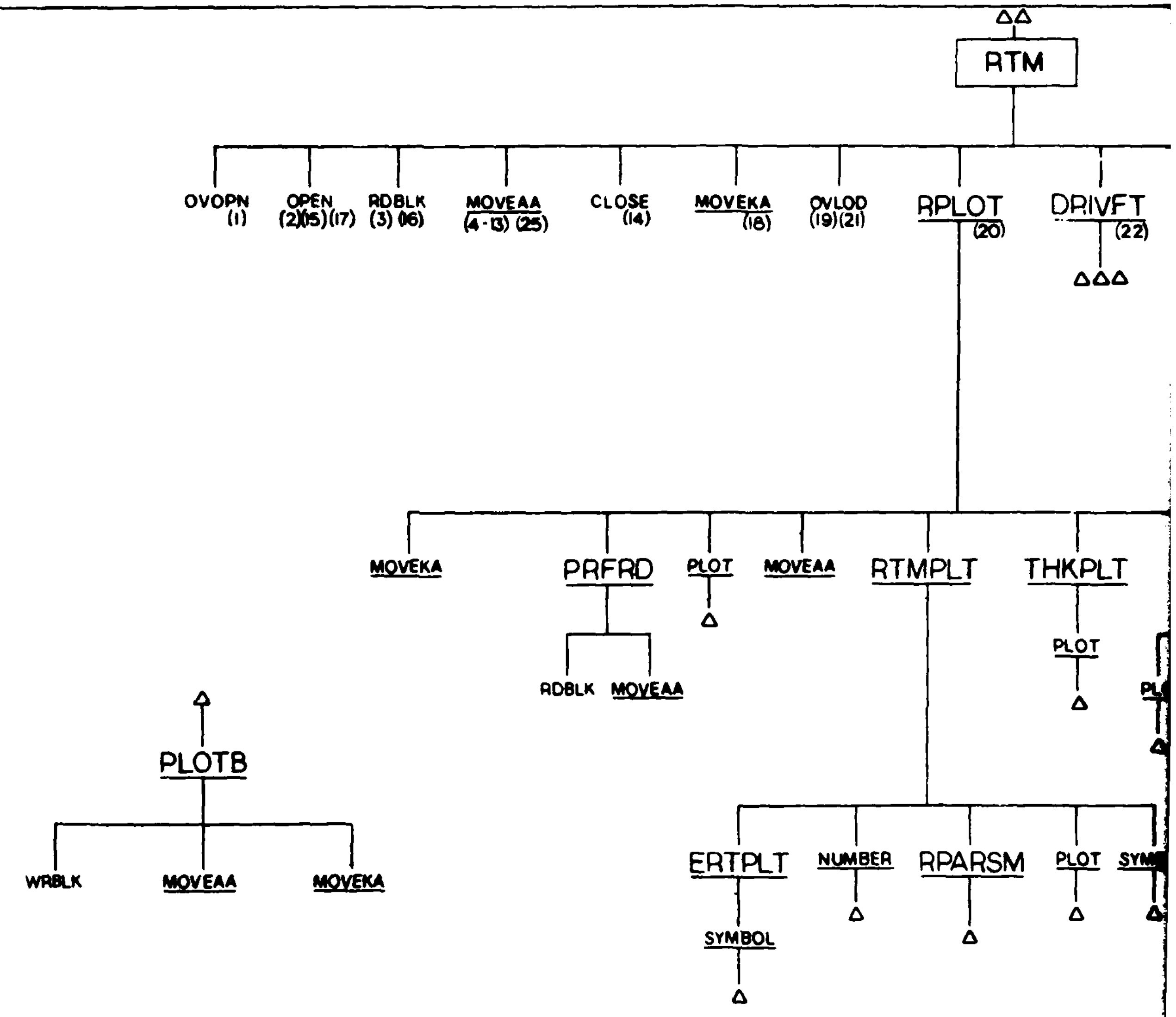
PERSP.SV

SUBROUTINE #	FUNCTION
1-2	OPEN AND READ SWAP FILE
3-13	MOVE HEADER INFO INTO C
14-15	OPEN, READ HEADER BLOCK
16	CHECK IF FEATURE DATA,
17	PERFORM PLOTTING
18	IF DESIRED, SWAP TO FTMAIN
19-22	FINISH PLOTTING AND SWA
23-24	RETURN TO MAIN PROGRA

7



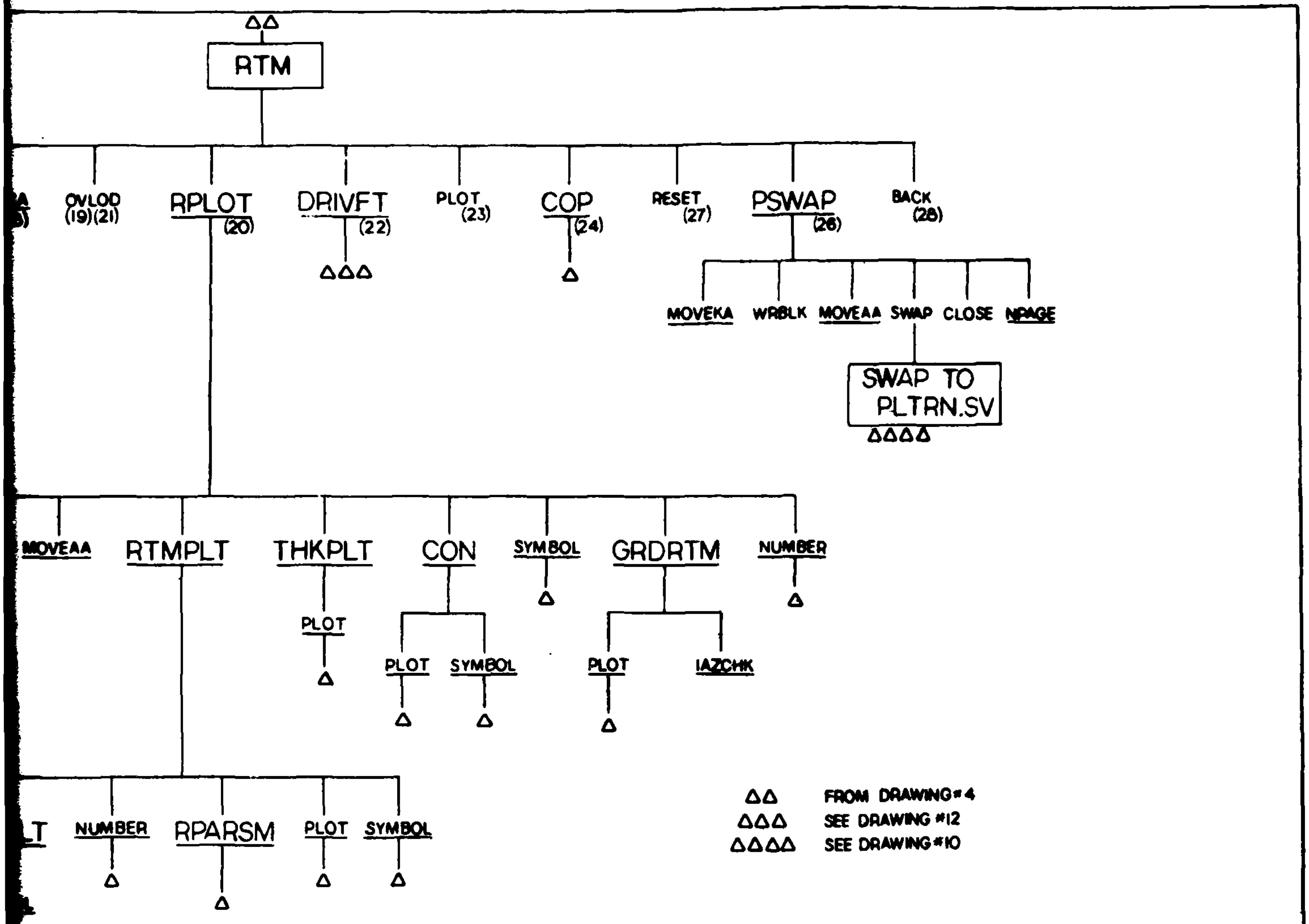
SUBROUTINE #	FUNCTION
1-2	OPEN AND READ SWAP FILE
3-13	MOVE HEADER INFO INTO COMMON BLOCKS
14-15	OPEN, READ HEADER BLOCK OF PROFILE FILE
16	CHECK IF FEATURE DATA, GRID LINES TO BE PLOTTED
17	PERFORM PLOTTING
18	IF DESIRED, SWAP TO FTMAIN.SV
19-22	FINISH PLOTTING AND SWAP TO WPLTRN.SV
23-24	RETURN TO MAIN PROGRAM (PLOAD.SV)



SUBROUTINE #	FUNCTION
1	OPEN OVERLAY FILE
2-3	OPEN AND READ SWAP FI
4-14	MOVE HEADER INFO. INTO
15-16	OPEN, READ HEADER BLOCK
17-20	PERFORM PLOTTING
21-22	PERFORM FEATURE DATA
23-26	FINISH PLOTTING AND SWA
27-28	RETURN TO MAIN PROGRA

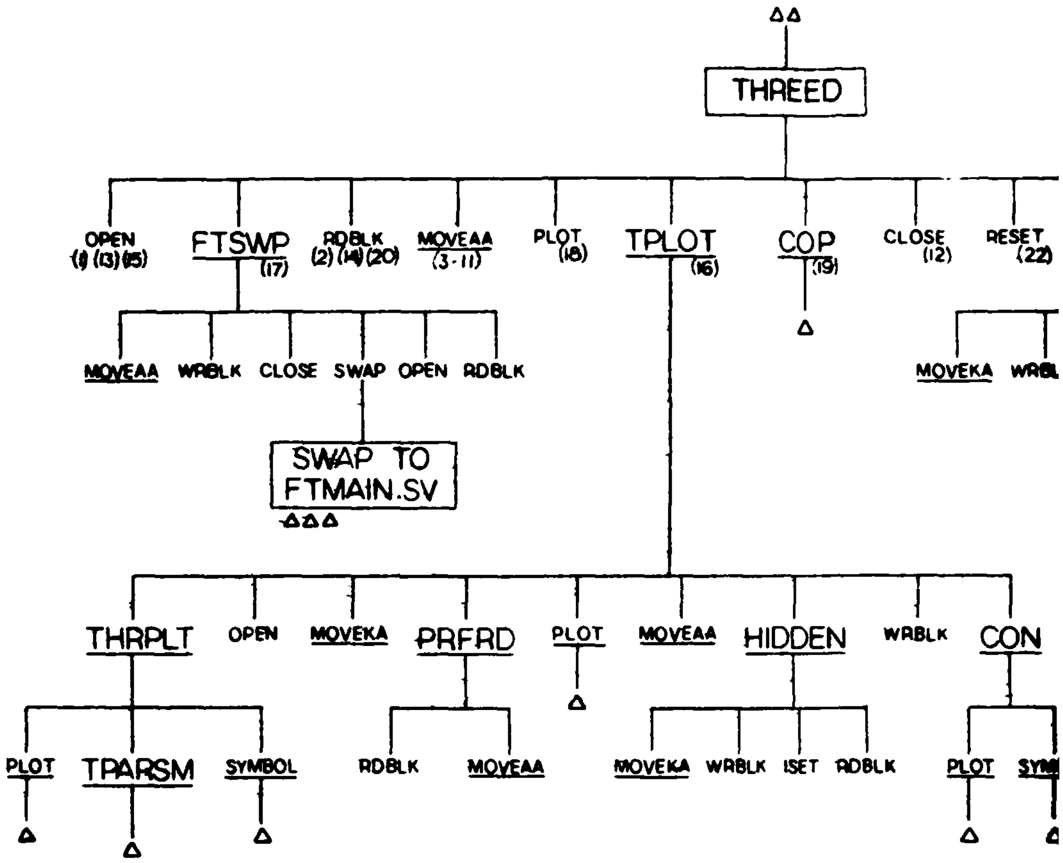
RTM.SV

8



UBROUTINE #	FUNCTION	OVERLAY USED (WRTM.OL)
1	OPEN OVERLAY FILE	
2-3	OPEN AND READ SWAP FILE	
4-14	MOVE HEADER INFO. INTO COMMON BLOCKS	
15-16	OPEN, READ HEADER BLOCK OF PROFILE FILE	
17-20	PERFORM PLOTTING	← RPTOL
21-22	PERFORM FEATURE DATA PLOTTING	← FETOL
23-26	FINISH PLOTTING AND SWAP TO WPLTRN.SV	
27-28	RETURN TO MAIN PROGRAM (RLOAD.SV)	

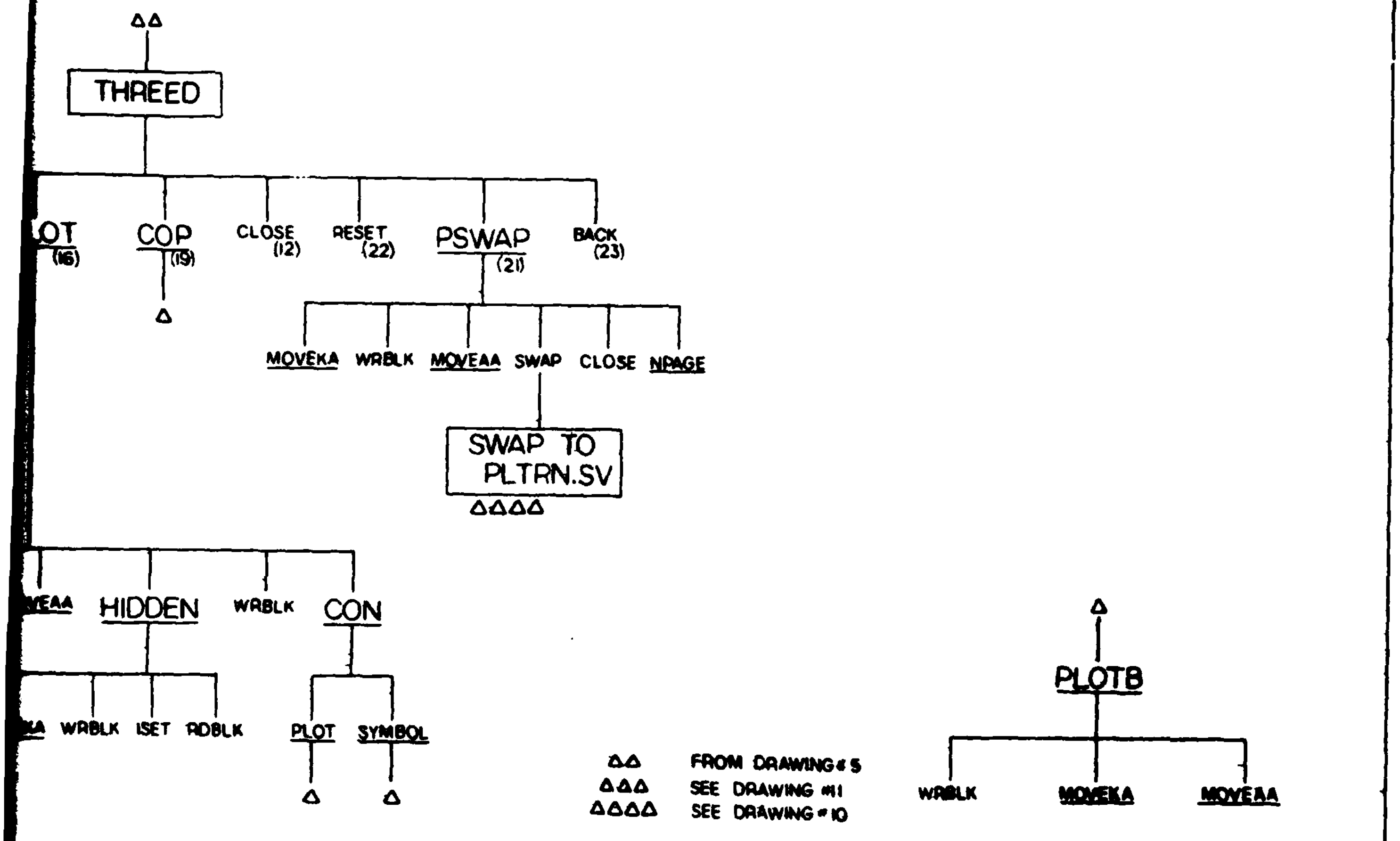
2



THREED.SV

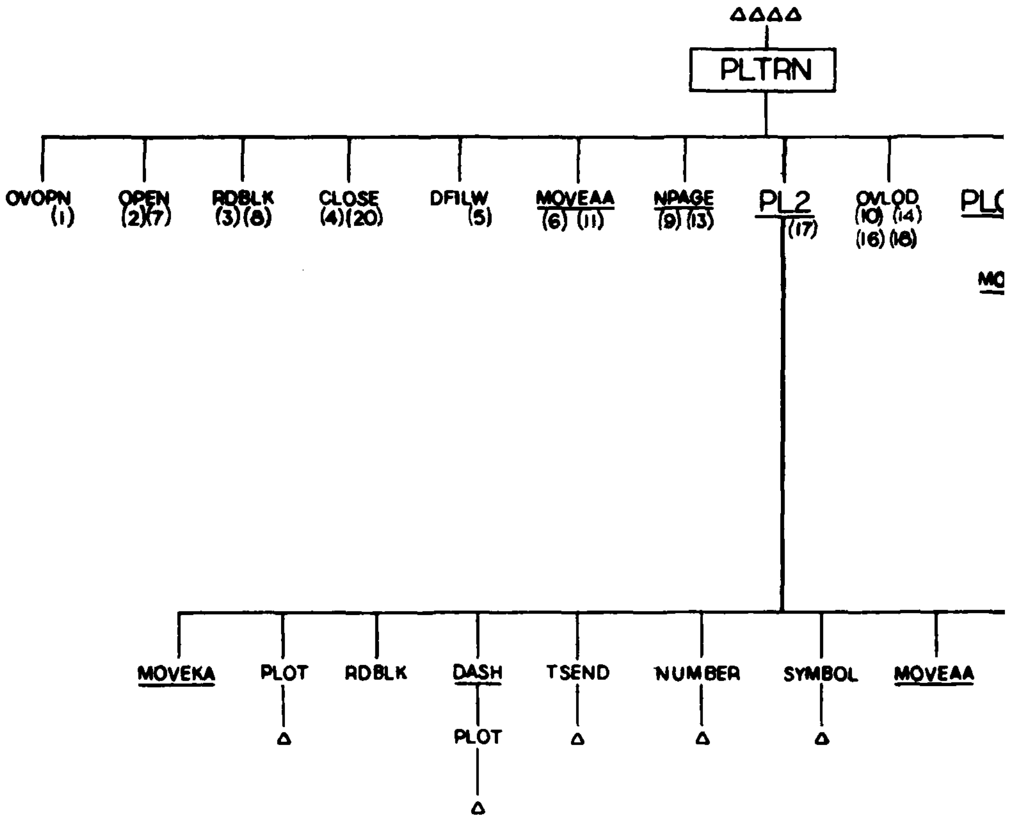
9

SUBROUTINE #	FUNCTION
1-2	OPEN AND READ SWAP F
3-12	MOVE HEADER INFO INT
13-14	OPEN, READ HEADER BLO
15	CHECK IF FEATURE LINE
16	PERFORM PLOTTING
17	IF DESIRED SWAP TO
18-21	FINISH PLOTTING AND S
22-23	RETURN TO MAIN PROG



BROUTINE #	FUNCTION
1-2	OPEN AND READ SWAP FILE
3-12	MOVE HEADER INFO INTO COMMON BLOCKS
13-14	OPEN, READ HEADER BLOCK OF PROFILE FILE
15	CHECK IF FEATURE LINES, GRID DATA TO BE PLOTTED
16	PERFORM PLOTTING
17	IF DESIRED SWAP TO FTMAIN.SV
18-21	FINISH PLOTTING AND SWAP TO WPLTRN.SV
22-23	RETURN TO MAIN PROGRAM (TLOAD.SV)

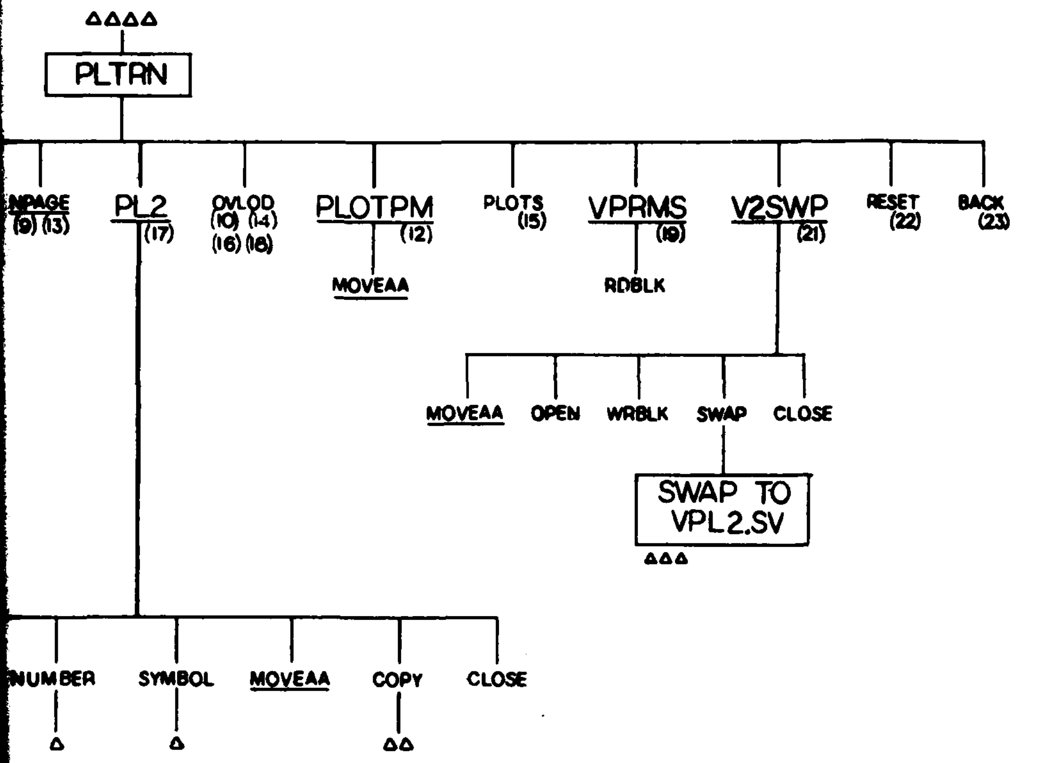
2



SUBROUTINE #	FUNCTION
1	INITIALIZE AND ENTER OU
2-3	READ HEADER OF "PLOTF'
4-6	CLOSE AND DELETE "P ENTER NAME FOR PE
7-13	READ HEADER OF PERMA
14-17	DO PLOTTING ON TEKT
18-21	SWAP TO VERSATEC PLOT
22-23	RETURN TO CALLING PR

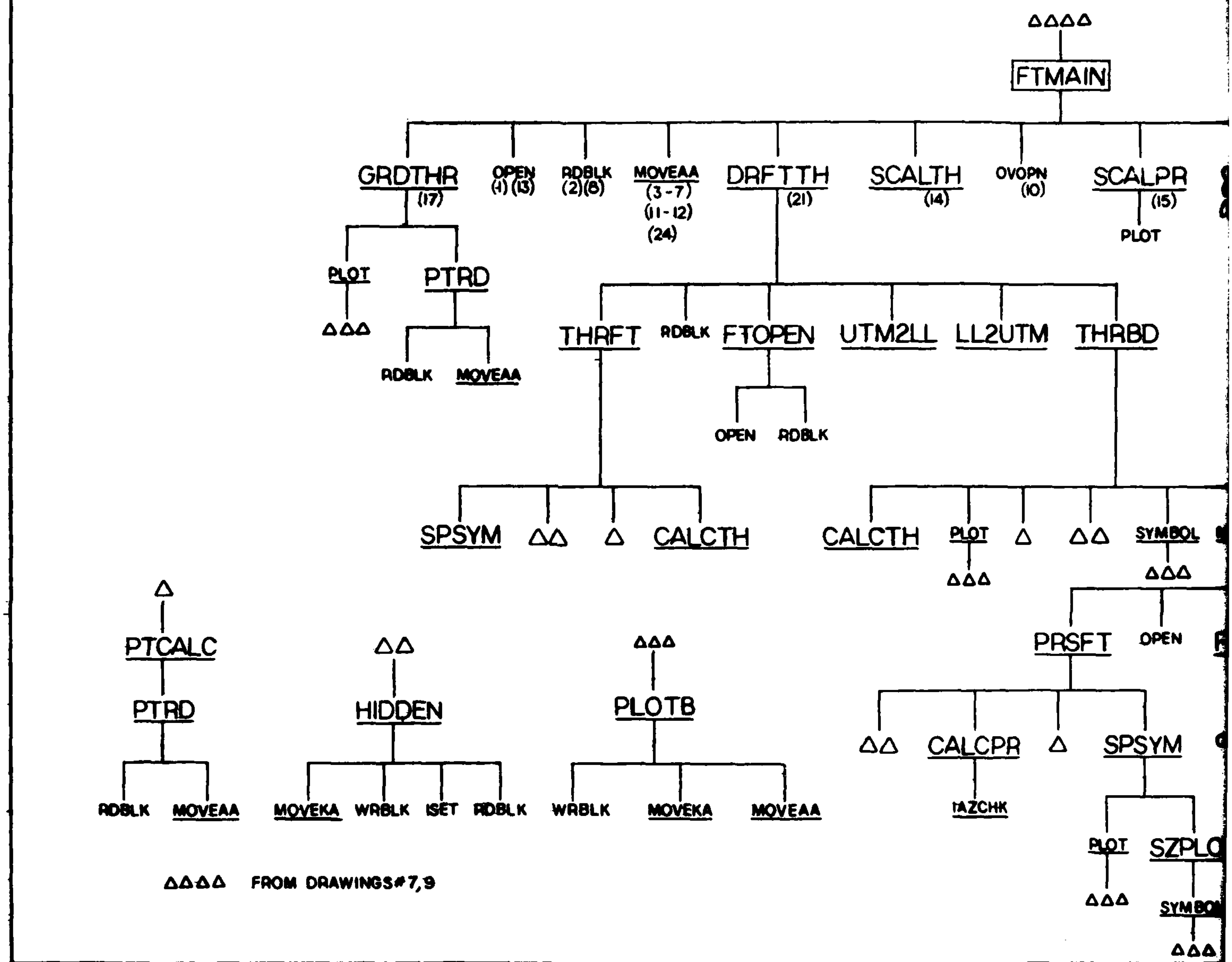
PLTRN.SV

10



Δ SEE DRAWING# 15
ΔΔ SEE DRAWING# 16
ΔΔΔ SEE DRAWING# 14
ΔΔΔΔ FROM DRAWINGS# 1,2,7,8,9

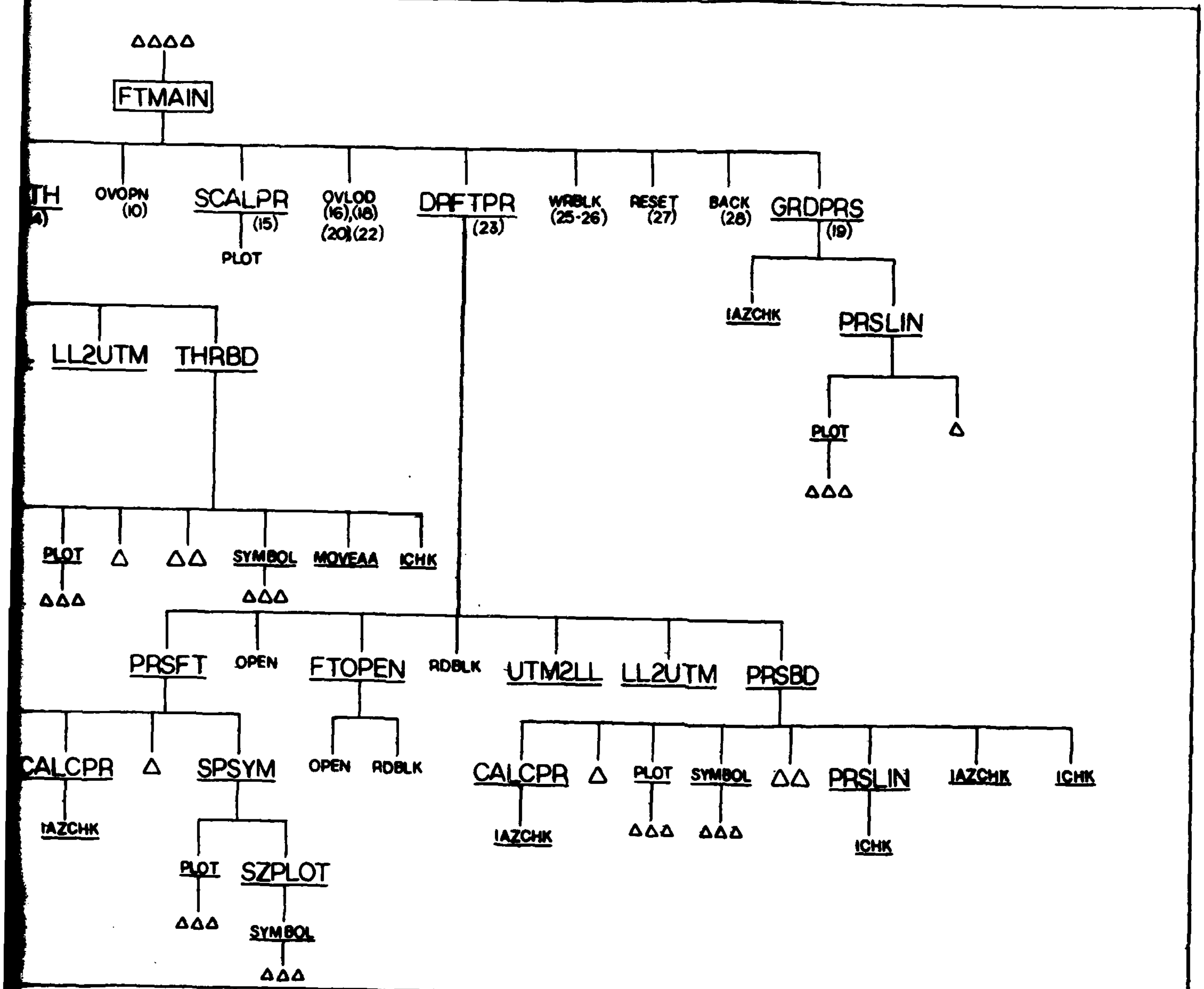
E#	FUNCTION	OVERLAY USED (WPLTRN.OL)
	INITIALIZE AND ENTER OUTPUT OPTION	
	READ HEADER OF "PLOTF" (TEMPORARY PLOT FILE)	
	CLOSE AND DELETE "PLOTF" AND ENTER NAME FOR PERMANENT PLOT FILE	
	READ HEADER OF PERMANENT PLOT FILE	← PLPOL
	DO PLOTTING ON TEKTRONICS SCREEN	← PLTOL, PL2OL
	SWAP TO VERSATEC PLOTTING ROUTINE, VPL2.SV	← VPROL
	RETURN TO CALLING PROGRAM	



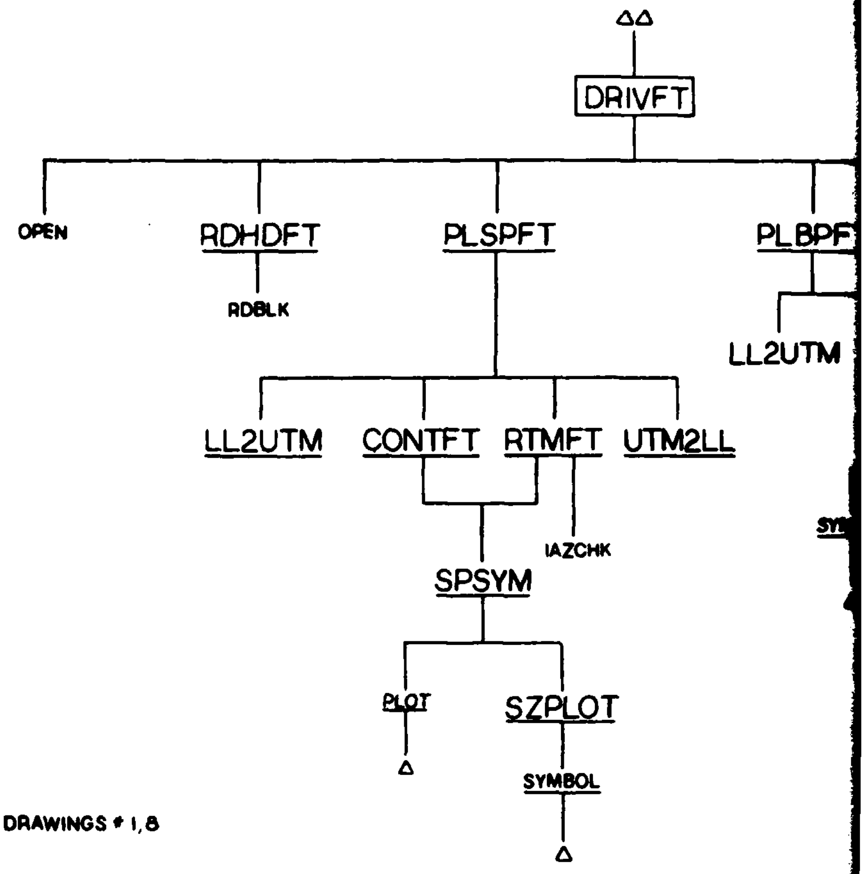
FTMAIN.SV

SUBROUTINE #	FUNCTION
1-2	OPEN POINT FILE
3-8	FILL COMMON BLOCKS
10-12	OPEN OVERLAY FILE AND
13	OPEN PLOT BUFFER FILE
14-15	DETERMINE PLOT CONST
16-19	IF DESIRED, PLOT GRID
20-23	IF DESIRED, PLOT FEATU
24-28	RETURN TO CALLING PR

11



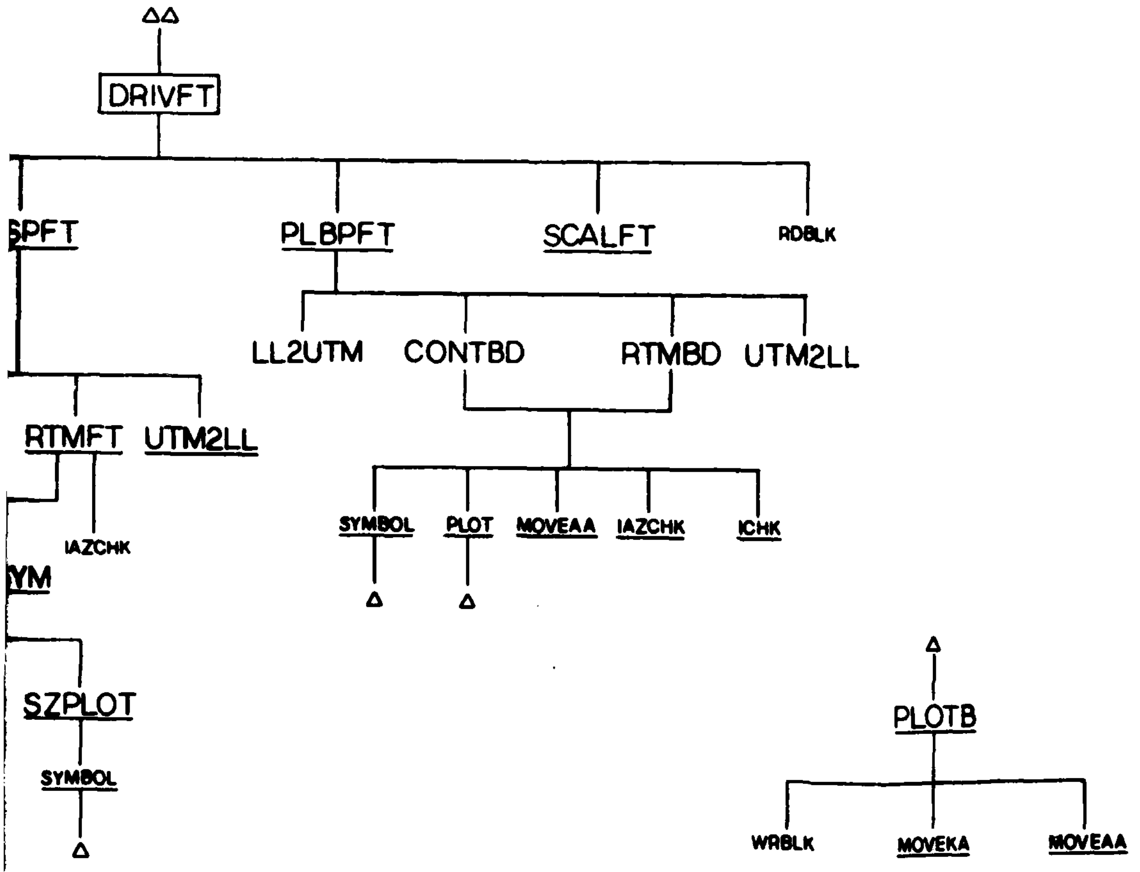
#	FUNCTION	OVERLAY USED (FTMAIN.OL)
	OPEN POINT FILE	
	FILL COMMON BLOCKS	
	OPEN OVERLAY FILE AND LOAD OVERLAY	
	OPEN PLOT BUFFER FILE	
	DETERMINE PLOT CONSTANTS	
	IF DESIRED, PLOT GRID	GTHOL OR GPROL
	IF DESIRED, PLOT FEATURE DATA	FTHOL OR FPROL
	RETURN TO CALLING PROGRAM	



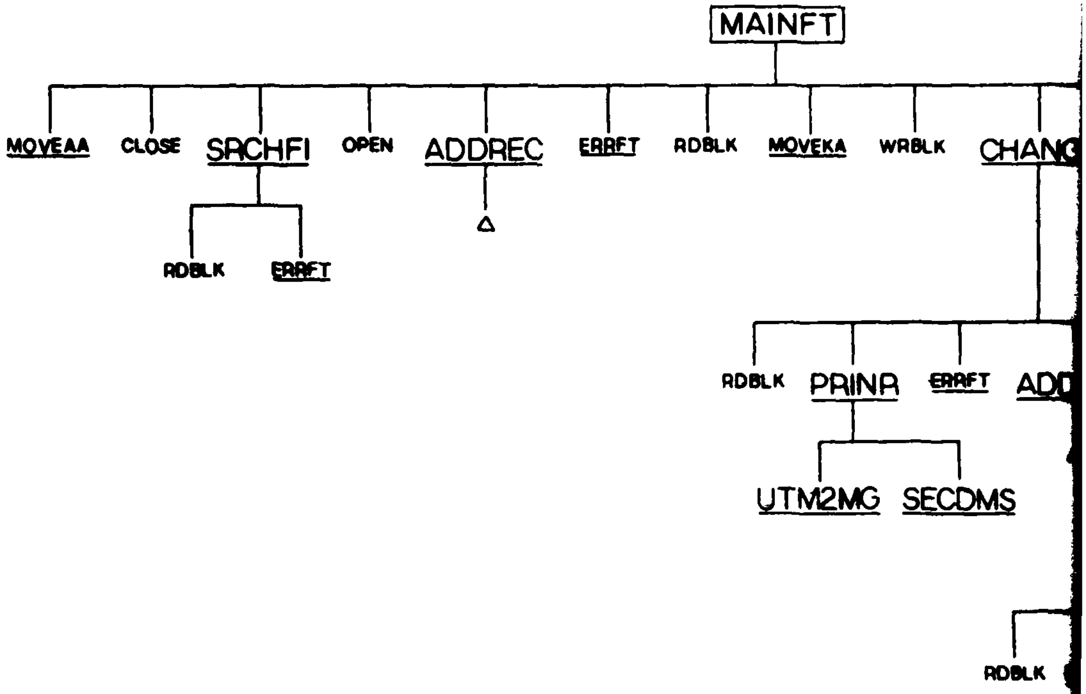
ΔΔ FROM DRAWINGS # 1,8

DRIVFT

12



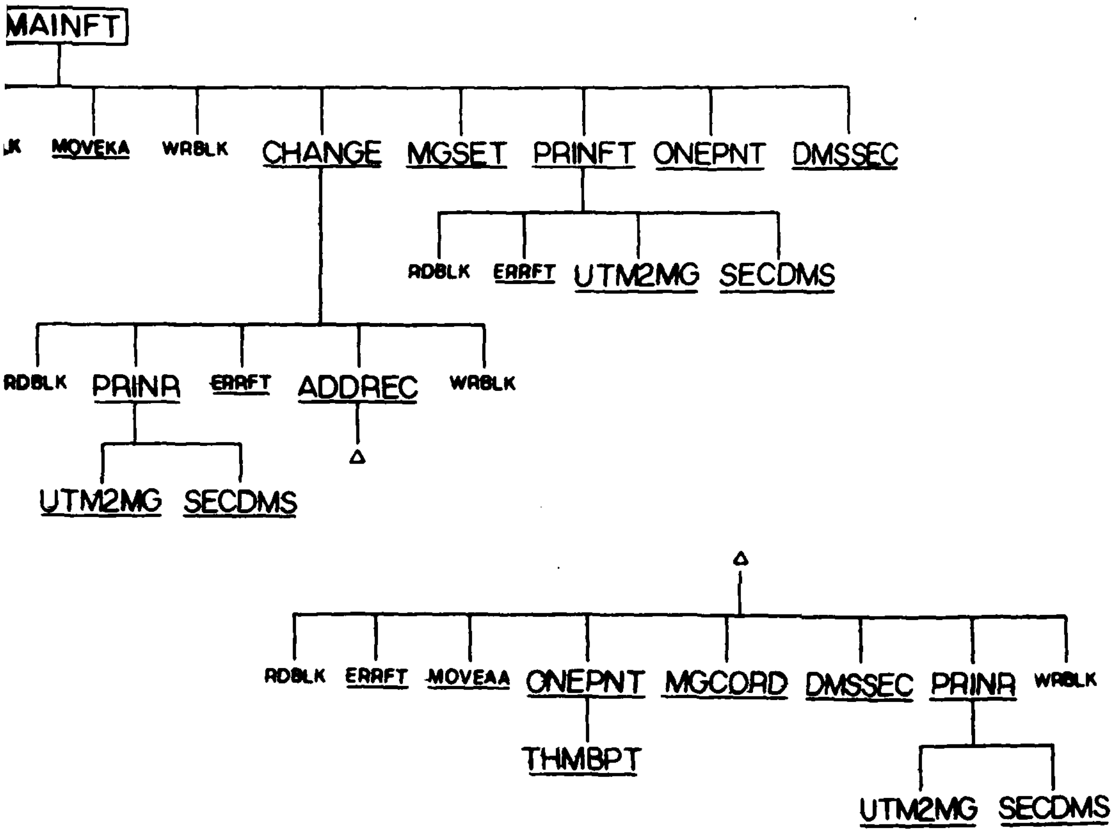
DRIVFT
SPFT
PLBPFT
SCALFT
RDBLK
LL2UTM
CONTBD
RTMBD
UTM2LL
RTMFT
UTM2LL
SYMBOL
PLOT
MOVEAA
IAZCHK
ICHK
IAZCHK
YM
SZPLOT
SYMBOL
PLOTB
WRBLK
MOVEKA
MOVEAA

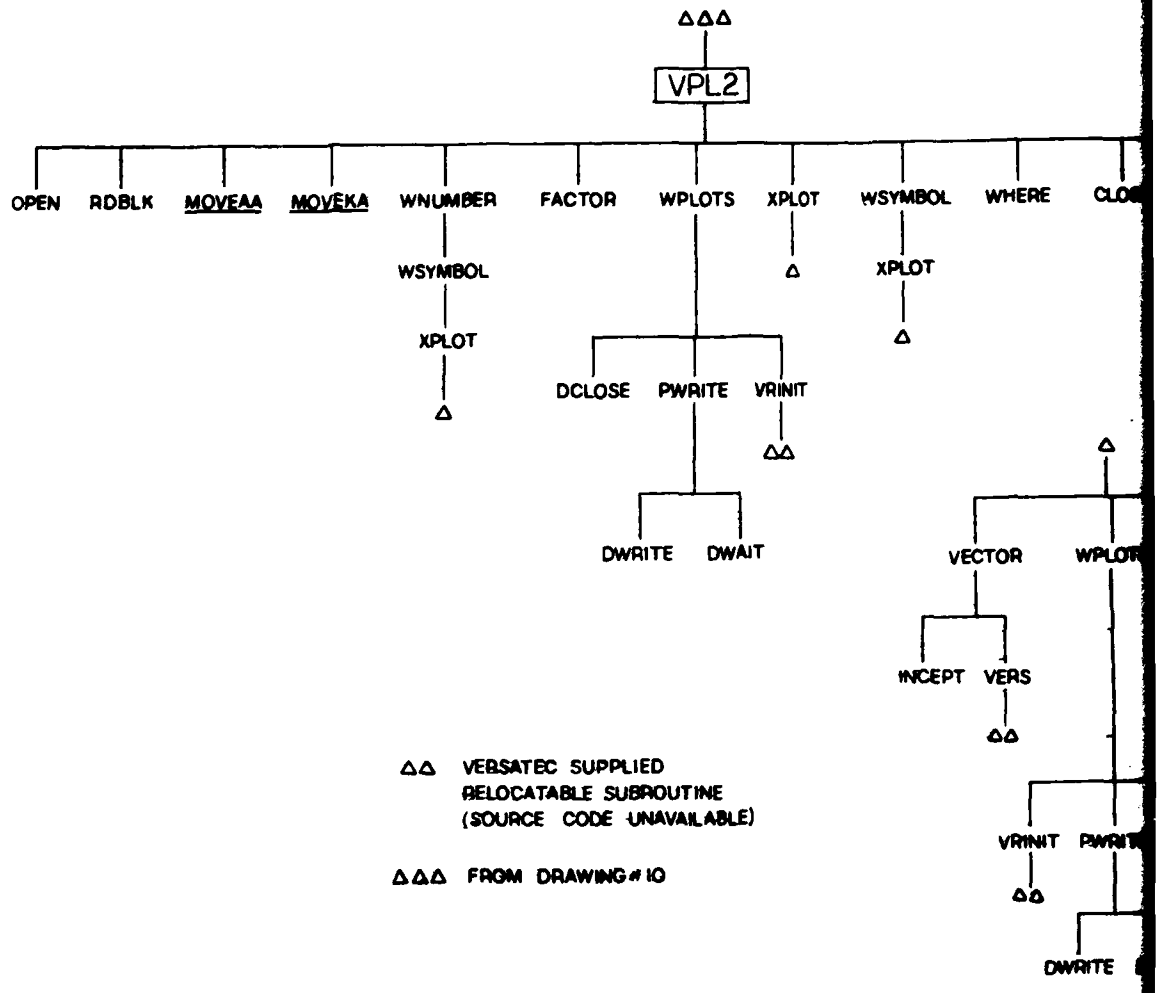


MAINFT.SV

13

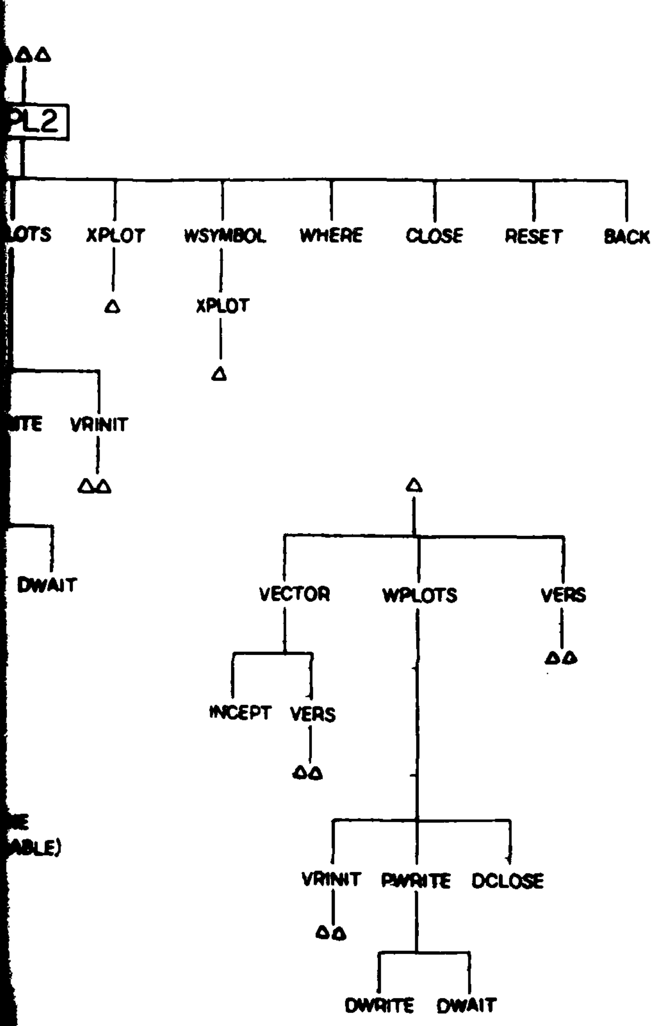
1


MAINFT
LK
MOVEKA
WRBLK
CHANGE
MGSET
PRINFT
ONEPNT
DMSSEC
RDBLK
ERRFT
UTM2MG
SECDMS
RDBLK
PRINR
ERRFT
ADDREC
WRBLK
UTM2MG
SECDMS
Δ
Δ
RDBLK
ERRFT
MOVEAA
ONEPNT
MGCORD
DMSSEC
PRINR
WRBLK
THMBPT
UTM2MG
SECDMS

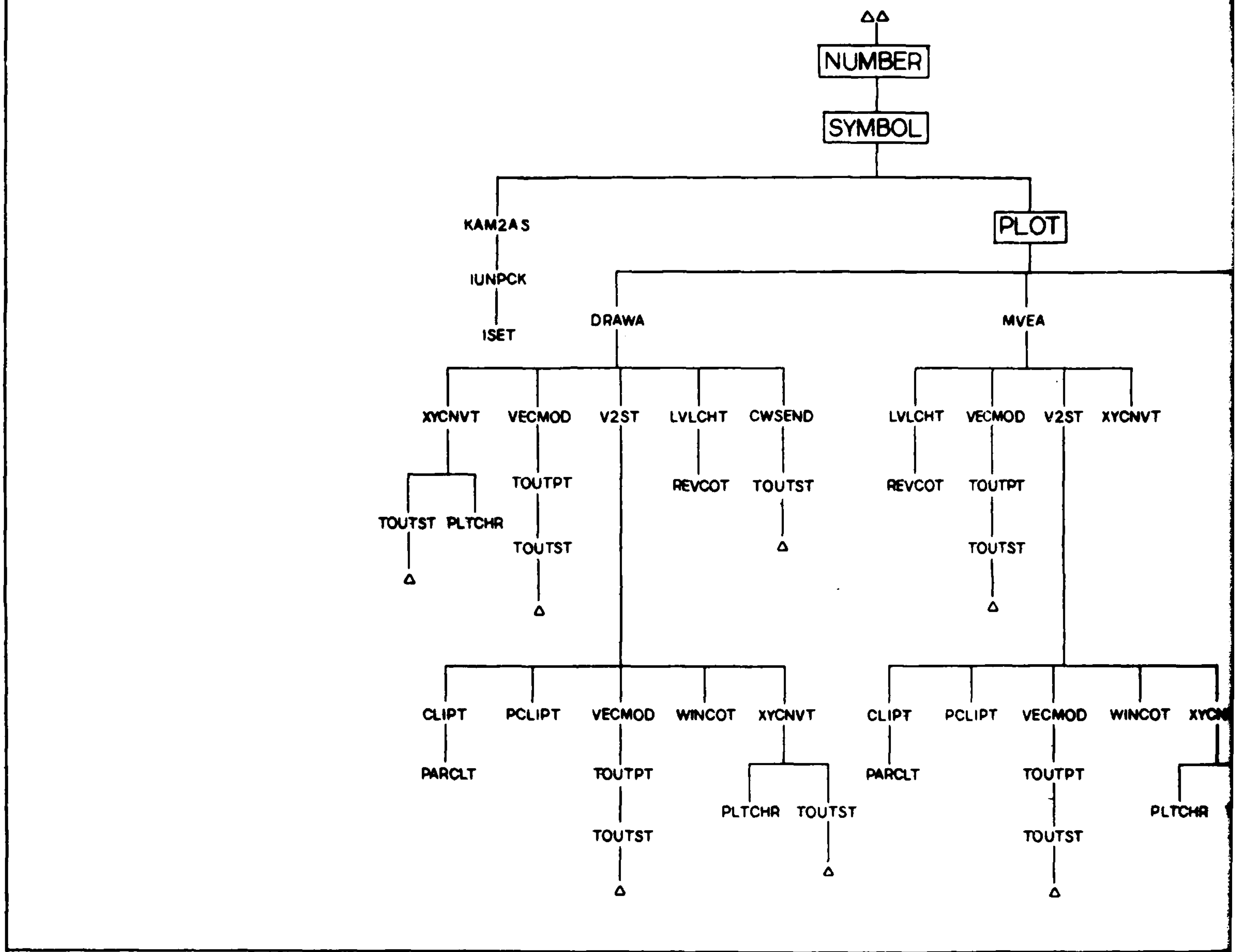

ΔΔΔ
VPL2
OPEN
RDBLK
MOVEAA
MOVEKA
WNUMBER
FACTOR
WPLOTS
XPLOT
WSYMBOL
WHERE
CLO
WSYMBOL
XPLOT
Δ
Δ
XPLOT
Δ
DCLOSE
PWRITE
VRINIT
ΔΔ
DWRITE
DWAIT
Δ
VECTOR
WPLOT
INCEPT
VERS
ΔΔ
VRINIT
PWRI
ΔΔ
DWRITE
ΔΔ VERSATEC SUPPLIED RELOCATABLE SUBROUTINE (SOURCE CODE UNAVAILABLE)
ΔΔΔ FROM DRAWING # 10

VPL2.SV

14

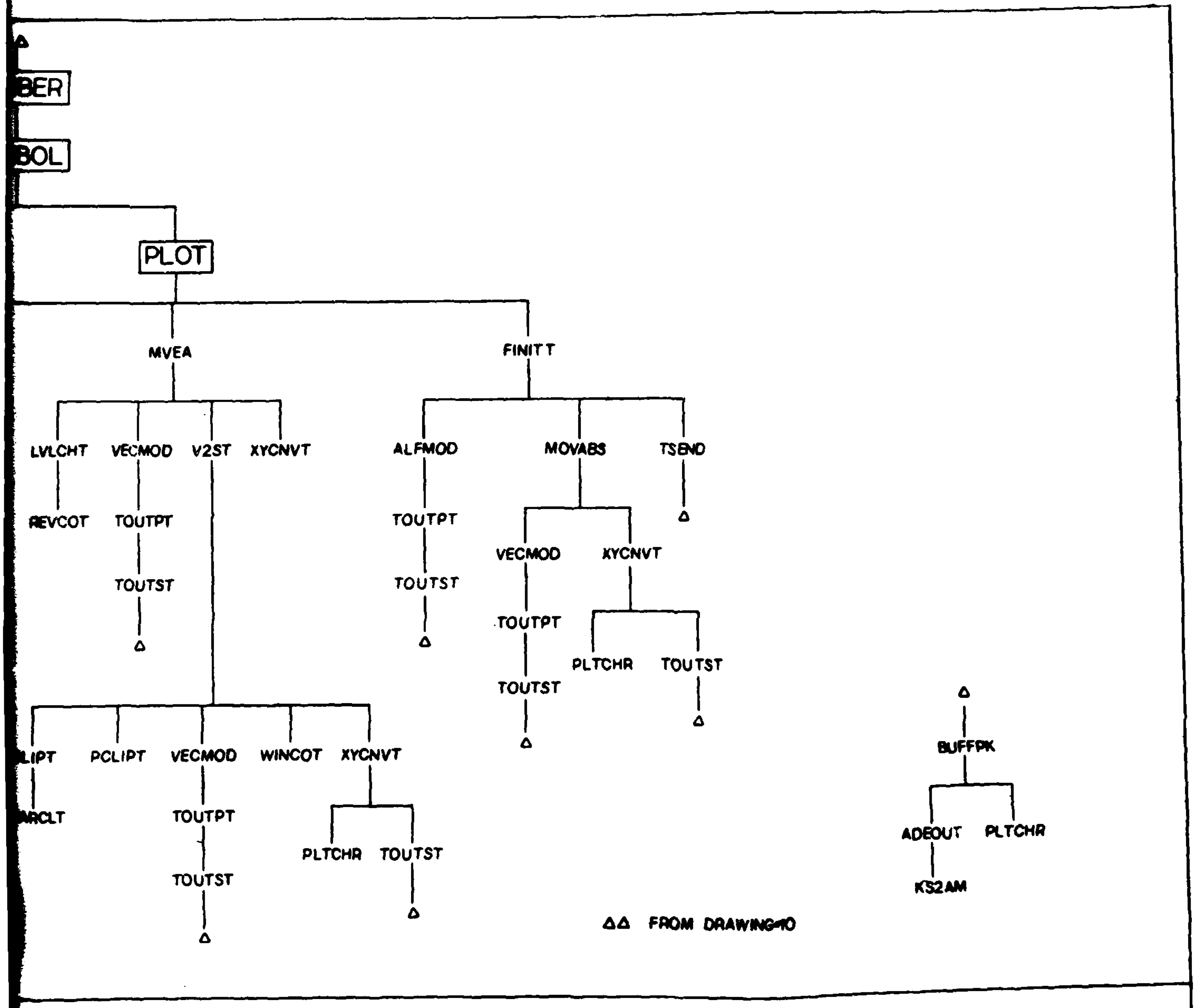


ΔΔ
PL2
LOTS
XPLOT
WSYMBOL
WHERE
CLOSE
RESET
BACK
Δ
XPLOT
Δ
ITE
VRINIT
ΔΔ
DWAIT
Δ
VECTOR
WPLOTS
VERS
ΔΔ
INCEPT
VERS
ΔΔ
NE
ABLE)
VRINIT
PWRITE
DCLOSE
ΔΔ
DWRITE
DWAIT



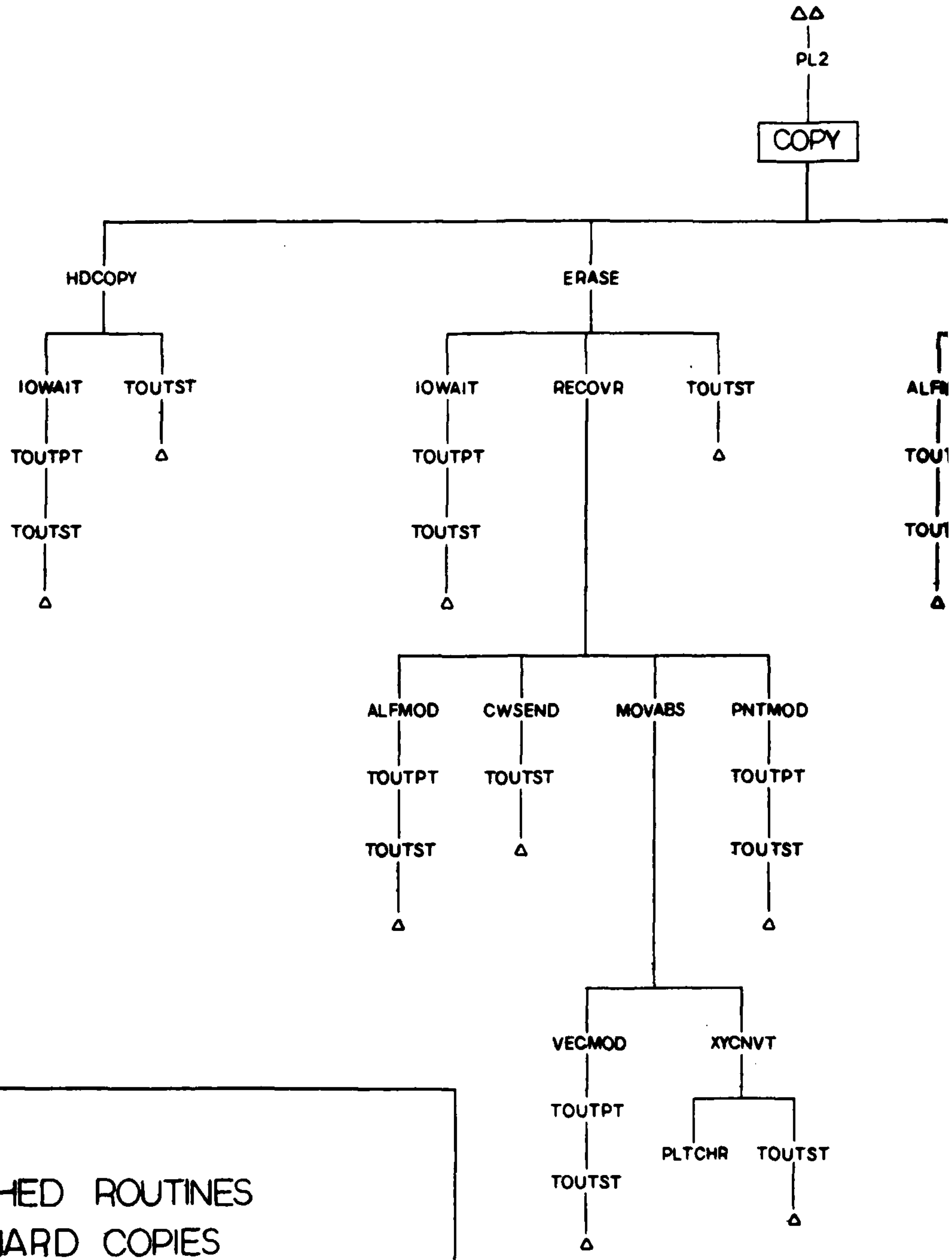
TEKTRONICS FURNISHED ROUTINES
NUMBER / SYMBOL /
PLOT

15



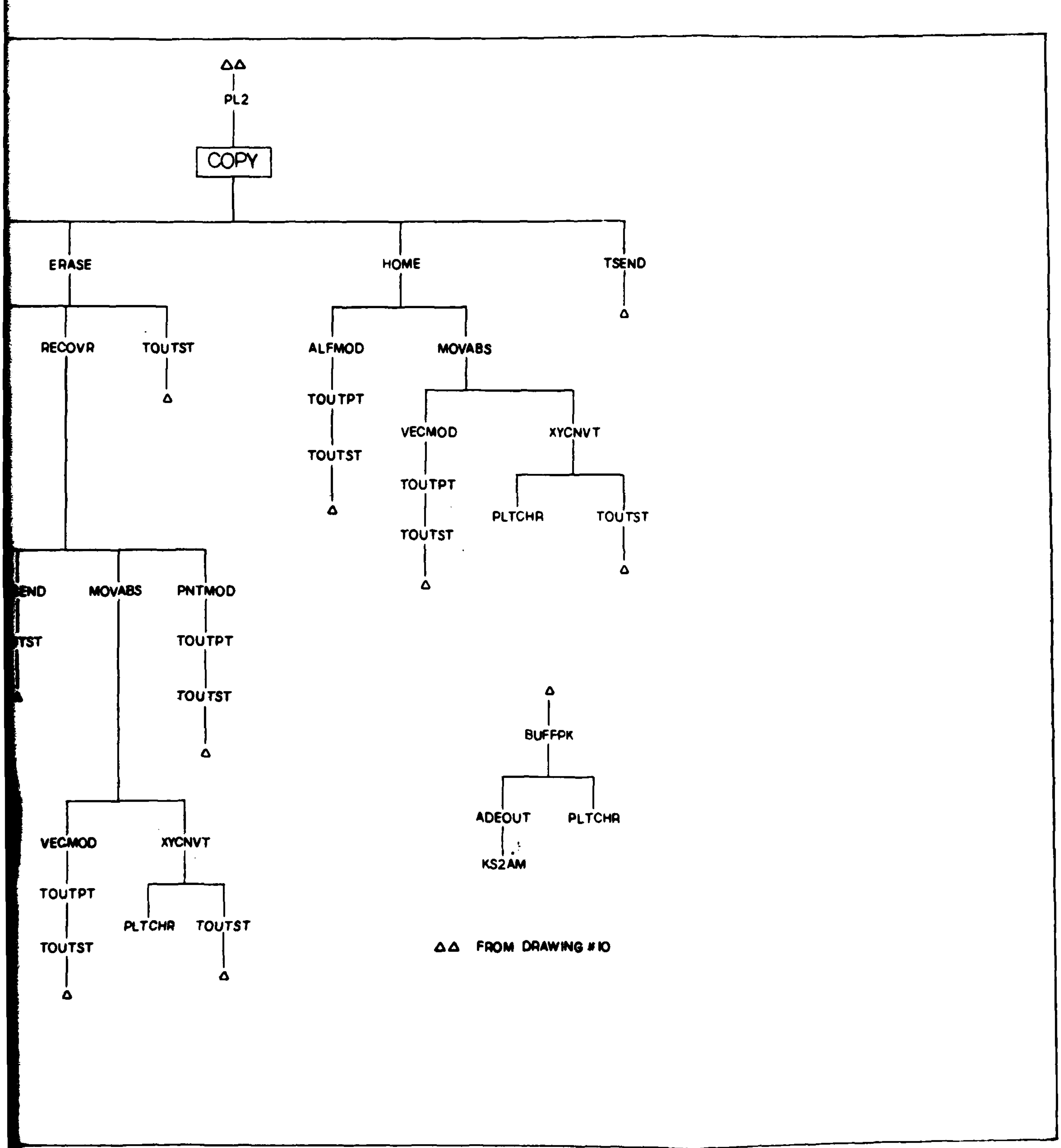
BER
BOL
PLOT
MVEA
FINITT
LVLCHT
VECMOD
V2ST
XYCNVT
ALFMOD
MOVABS
TSEND
REVCOT
TOUTPT
TOUTST
TOUTPT
TOUTST
VECMOD
XYCNVT
TOUTPT
TOUTST
PLTCHR
TOUTST
LIPT
PCLIPT
VECMOD
WINCOT
XYCNVT
RCLT
TOUTPT
TOUTST
PLTCHR
TOUTST
BUFFPK
ADEOUT
PLTCHR
KS2AM
ΔΔ FROM DRAWING-10

2



TEKTRONICS FURNISHED ROUTINES
FOR MULTIPLE HARD COPIES

16


ΔΔ
PL2
COPY
ERASE
HOME
TSEND
Δ
RECOVR
TOUTST
Δ
ALFMOD
TOUTPT
TOUTST
Δ
MOVABS
VECMOD
TOUTPT
TOUTST
Δ
XYCNVT
PLTCHR
TOUTST
Δ
END
TST
MOVABS
PNTMOD
TOUTPT
TOUTST
Δ
VECMOD
TOUTPT
TOUTST
Δ
XYCNVT
PLTCHR
TOUTST
Δ
Δ
BUFFPK
ADEOUT
KS2AM
PLTCHR
ΔΔ FROM DRAWING #10

2

III. DISK FILE I/O

IDENTIFIER	FILE TYPE	PRIMARILY USED BY:
- .DB	Data Base files	(C,P,R,L,T)LOAD
- .FD	Feature Data files	FTMAIN,MAINFT,DRIVFT
- .PF	Profile files	PROFL
- .PL	Plot files	PLTRN
- .PM	Parameter files	DOPARM,PLOTPM
"PNTFL"	Hidden Point File	PERSP,THREED,FTMAIN
"SWPFL"	Swap file	ALL
"SYMFL"	Tektronix Symbol file	PLTRN
"V2SFL"	Versatec Swap file	VPL2

IIIA. FILES, USES, LOGICAL UNITS

Unit No. and File Name	Opened	Closed	Read From	Written To
22,3	PTMN	SETUP	PTMN block 0	NONE
DATA BASE	SETUP		NALT	
FILE	PROFL		WINDOW	
.DB			SETUP block 0	
			NALT	
1,26	MAINFT	MAINFT	MAINFT	MAINFT
.FD	DRIVFT		ADDREC	ADDREC
FEATURE	DRFTPR		CHANGE	CHANGE
DATA	FFTOPEN		PRINFT	
FILE			SRCHFI	
			DRFTTH	
			DRIVFT	
			RDHDFT block 0	
			FTOPEN block 0	
			DRFTPR	
21	PERSP	CPLOT	PERSP block 0	PTMN block 0
PROFILE	PROFL	PRSWP	RTM	PRFWRT
FILE	RTM	RTSWP	THREED block 0	
.PF	THREED	LMAIN	PRFRD	
	JPRFRT	THSWP	PRFRD	
	WPRFRT	PTMN	JPRFRT block 0	
		PRFRT	WFRFRT block 0	

Unit No. and File Name	Opened	Closed	Read From	Written To
32 "PLOTF" .PL PLOT FILE	CPLOT PLTRN FTMAIN VPL2 PLTRN RTM TPLOT FTSWP LOSPLT PPLOT	VPL2 PLTRN FTSWP PL2 PSWAP "PLOTF" deleted or renamed in PLTSV	VPL2 PLTRN block 0 PL2 VPRMS	PLOTB PSWAP
44 "PLPARM" .PM PARAMETER FILE	PLOTPM INCON INLOS INPRS INRTM INTHRD PMPRT	PLOTPM INCON INLOS INRTM INTHRD PMPRT WPMPRT "PLPARM" deleted or renamed in PLTSV	PLOTPM PMPRT WPRFRT	INCON INLOS INPRS INRTM INTHRD PMPRT PRFRT

Unit No. and File name	Opened	Closed	Read From	Written To
33	PERSP	FTSWP	FTMAIN	FTMAINI
"PNTFL"	FTMAIN		PTRD	FTSWP
POINT	THREED		FTSWP	HIDDEN
FILE	FTSWP		HIDDEN	PPLOT
				TPLOT
38,23	PRSWP	PRSWP	PERSP	PRSWP
"SWPFL"	RTSWP	RTSWP	RTM	RTSWP
SWAP	THSWP	THSWP	THREED	THSWP
FILE	PERSP	PERSP	PROFL	JPRFRT
	RTM	RTM		WPRFRT
	THREED	THREED		
	PROFL	JPRFRT		
	JPRFRT	WPRFRT		
	WPRFRT			
25	SYMBOL	SYMBOL	SYMBOL	NONE
"SYMFL"				
TEKTRONIX				
SYMBOL				
FILE				
42	VPL2	NONE	VPL2 block 0	V2SWP block
"V2SFL"				
VERSATEC				
SWAP				
FILE				

IIIB. DESCRIPTIONS OF FILE STRUCTURE

SWAP FILE
"SWPFL"

LUN: 38,23

READ BY: PROFL,PERSP,RTM,THREED

WRITTEN TO: JPRFRT,WPRFRT,PRSWP,RTSWP,THSWP

PURPOSE: File is loaded up with all the necessary parameters that need to be passed between main programs that SWAP to other executables for further processing.

STRUCTURE: Swap between (CLOAD,PLOAD,RLOAD,LLOAD,TLOAD) and PROFL

Words	Variable Type	Contains
1-4	INTEGER	File name of profile file (.PF)
5-8	INTEGER	File name of data base file (.DB)
9-42	INTEGER	IPRF(32) - see common PRFCM. This area is filled with data from previous .PF file if used.
43	INTEGER	IDATA - see common UNITS
44	INTEGER	INPUT - see common UNITS
45	INTEGER	INCOV - see common UNITS
46	INTEGER	ISPHER - see common PARAM
47	INTEGER	IH - see common PARAM
48	INTEGER	IZONE - see common PARAM
49-50	REAL	REAST - see common PARAM
51-52	REAL	RNORTH - see common PARAM

Swap between PLOAD and WPERSP, RLOAD and WRTM

Words	Contains
1-4	Profile file name
5-8	Data base file name
9-10	Common PNTS
11-12	Common FAC
13-15	Common EARTH
16	Common GRDBLK
17	Common RIDGE
18-40	Common CONT
41-70	Common PRSBLK,RTMBLK,THRBLK
71-74	Common UNITS
75-80	Common PARAM
81-85	Common FDFIL
86-89	Common LOCUTM

DATA BASE FILES
(.DB)

LUN: 22

OPENED BY:	SETUP,PTMN
CLOSED BY:	SETUP
READ BY:	PTMN,NALT,ALT,WINDOW,SETUP
WRITTEN TO:	NONE

PURPOSE: Primary input file for FEED programs. Contains the topographic data for a geographic region.

STRUCTURE: Header Block-256 words

Words	Variable Type	Contains
1-2	REAL	Southern Most Latitude (in seconds) (Pos = N) (neg = S)
3-4	REAL	Northern Most Latitude (in seconds) (Pos = N) (neg = S)
5-6	REAL	Western Most Longitude (in seconds) (Pos = E) (neg = W)
7-8	REAL	Eastern Most longitude (in seconds) (Pos = E) (neg = W)
9	INTEGER	X Cell size (in seconds)
10	INTEGER	Y Cell Size (in seconds)
11	INTEGER	Number of cells per row

12	INTEGER	Number of rows
13	INTEGER	Number of bytes per block
14	INTEGER	Start position of data for region I.
15-29		Identifying information for Region 2 (same as words 1-14)
30-44		Identifying information for Region 3
45-59		Identifying information for Region 4
60-74		Identifying information for Region 5
'		'
'		'
'		'
253	INTEGER	Indicates whether data base coordinates are WGS or UTM.
254-256		Empty

Body of file follows:

257	INTEGER	Elevation for Cell 1,1
258	INTEGER	Elevation for Cell 1,2
259	INTEGER	Elevation for Cell 1,3
"	"	"
"	"	"
etc.	etc.	etc.

USE: SETUP reads in the header block. Tests word 11 for length of data base (if 0, then file is empty) and word 253 for type of coordinates stored in header (1 = WGS, 2 = UTM)

PTMN reads in the header for the sole purpose of loading up COMMON/INDEX/MASTER. Although the integer array containing the header (MASTER) is equivalent to a real array XMAS, XMAS is never used in this program.

WINDOW reads a portion of the polynominal data base (up to 2000 words) into core so that further processing will be faster.

ALT reads in the elevations for lat/lon locations from polynomial data base.

NALT reads in the elevations for lat/lon location from gridded data base.

PROFILE FILE
(.PF)

LUN: 21

READ BY: PERSP,RTM,THREED,()RFRD,PRFRD,()RFRT,()RFRT
WRITTEN TO: PTMN,PRFWRT

PURPOSE: Contains the terrain profiles extracted from the data base by PROFL. The profiles are processed and plotted by each of the primary FEED programs.

STRUCTURE: Header Block - 256 bytes

Words	Contains
1-4	Profile File name
5-8	Data Base File Name
9-10	IALT,ICOUNT - see common PNTS
11	Empty
12	ITYPE - which program created this profile file - (1 = LLOAD, 2 = RLOAD, 3 = CLOAD, 4 = Unused, 5 = PLOAD, 6 = TLOAD)
13-27	Contents of commons LOSBLK,RTMBLK,CONBLK,PRSBLK, THRBLK (respectively depending on ITYPE) starting with the real boundary coordinates (usually RL or XL, after LDMS). Note - the length of these common blocks varies.
27-28	For LLOAD,CLOAD,PLOAD
29-30	For TLOAD Different position is caused by different lengtn noted above Word 1 is NBLK - tne number of 256 word blocks a profile would fill - (NPT - 1)/256 + 1 Word 2 is NBLK * NSCAN - see commons listed above
30-256	Empty

Body of file begins here:

257	INTEGER	Elevation of 1st point in profile 1
258	INTEGER	Elevation of 2nd piont on profile 1
259	INTEGER	Elevation of 3rd point on profile 1
260	INTEGER	Elevation of 4th point in profile 1
"	"	"
"	"	"
etc.	etc.	etc.
"	"	Elevation of last point on profile 1
"	"	Elevation of 1st point on profile 2
"	"	"
"	"	"
etc	etc	etc

FEATURE DATA FILE
(.FD)

LUN: 1,26

READ BY: MAINFT,ADDREC,CHANGE,PRINFT,SRCHFI,DRFTTH,DRIVFT,RDHDFT, FTOPEN,DRFTPR

WRITTEN TO: MAINFT,ADDREC,CHANGE

PURPOSE: This file holds the feature data (unit symbols, roads, railroads, urban areas, etc) that is to be overlayed onto the various FEED plots. The file is created and maintained by MAINFT.SV. The feature data is plotted by routines DRIVFT,DRFTTH, and DRFTPR.

STRUCTURE:

Words	Variable Type	Contains
1-4	INTEGER	File name of feature data file (.FD)
5	INTEGER	"W" if file coordinates are WGS "U" if file coordinates are UTM
6-15	INTEGER	20 character file description
16	INTEGER	Number of single point records in file
17	INTEGER	Number of boundary type records in file
18	INTEGER	Number of disk blocks in file
19	INTEGER	Total number of records in file

The feature data records follow:

20	INTEGER	Feature record number
21	INTEGER	Feature code 20 is point type data 20 is boundary type data
22-33	INTEGER	20 character record description
34	INTEGER	Number of points (coordinate pairs) in record
35-36	REAL	Latitude (or Northing) of point 1
37-38	REAL	Longitude (or Easting) of point 1
39-42	REAL	Coordinates for point 2
43-46	REAL	Coordinates for point 3
"	"	"
etc.	etc.	etc.

The feature data records repeat in sequence in the same format. If a record has too many points to fit in the remainder of the disk block, the block is padded with -1, and the record is written out into the next block. Thus records do not span block boundaries.

PARAMETER FILE
(.PM)
"PLPARM"

LUN: 44

READ BY: PLOTPM,PMPRT,WPRFRT

WRITTEN TO: INCON,INLOS,INPRS,INRTM,INTHRD,PMPRT,WPRFRT

PURPOSE: This file contains the parameters selected by the user which define the type of plot produced. Thus the parameters in "PLPARM" correspond to the plot in "PLOTF". If the user saves the plot under a unique name (XXXX.PL), then the matching parameter file (XXXX.PM) is created.

STRUCTURE:

Words	Variable Type	Contains
1	INTEGER	Plot type - 1 = contour 2 = line of sight 3 = 3-D 4 = perspective 5 = RTM
2-3	INTEGER	4 character data base file name
4	INTEGER	'Y' or 'N' - whether feature data is to be plotted
5-6	INTEGER	4 character feature data file name
7	INTEGER	INPUT - 1 = WGS 2 = UTM
9-10	REAL	Smallest possible UTM Easting (used if INPUT = 2)
11-12	REAL	Smallest possible UTM Northing (used if INPUT = 2)

The contents of the following word positions vary depending on the type of plot being generated:

Contour Plot:

13-28	INTEGER	Bounding Coordinates
29-30	REAL	Latitude spacing in seconds (or northing spacing in meters)
31-32	REAL	Longitude spacing in seconds (or easting spacing in meters)
46	INTEGER	Number of points between TIC marks

3-D Plot:

13-28	INTEGER	Bounding coordinates
29-30	REAL	Latitude interval
31-32	REAL	Longitude interval
33-35	INTEGER	Facing boundary
36-41	INTEGER	Position of viewer with respect to facing boundary
42	INTEGER	Reference elevation
49-50	REAL	Vertical exaggeration

Line of Sight Plot:

13-28	INTEGER	Beginning and enpoint coordinates
29	INTEGER	Number of points along the profile.
49	INTEGER	IPRT = 0 plot only 1 both plot and tabular printout 2 tabular printout only

Perspective Plot:

13-20	INTEGER	Coordinates of observation point
21-22	REAL	Height of observer
23-24	REAL	Bearing
25-26	REAL	Radial length
27-28	REAL	Spacing along radial
29-30	REAL	Spacing between radial
31-32	REAL	Field of view - (degrees)
49-50	REAL	Vertical exaggeration.

RTM Plot:

13	INTEGER	Masking option: 1 - safe area contours 2 - acquisition contours 3 - safe area below given ceiling 4 - fields of fire
14	INTEGER	ICH - cross hatching - 0 = yes

15-22	INTEGER	Coordinates of observer position
23	INTEGER	Height of observer
24	INTEGER	Ceiling height
25-26	REAL	Radius of coverage
27-28	REAL	Spacing along radials
29-30	REAL	Bearing of first radial
31-32	REAL	Bearing of last radial
33-34	REAL	Spacing between radials

Several of the plot types share the following variables:

43	INTEGER	ICR = 1 or 3 range lines will be plotted = 2 or 4 contour lines will be plotted = 0-4 grid lines will be plotted = 3-4 ridge lines will be plotted
44	INTEGER	Minimum contour level
45	INTEGER	Contour interval
46	INTEGER	Grid spacing

47	INTEGER	IRDGE - 0 - ridge lines will be plotted 1 - ridge lines will not be plotted
48	INTEGER	IECC - earth correction option selected by user
100-101	INTEGER	4 character profile name

All other file positions are used.

PLOT FILE
(.PL)
"PLOTF"

LUN: 32

READ FROM: PLTRN,PL2,VPL2,VPRMS
WRITTEN TO: PLOTB,PSWAP

PURPOSE: This file is used to store all the plot commands generated by the various FEED programs. The commands are then read from the file by PLTRN.SV and actually plotted on the Tektronix and/or Versatec.

STRUCTURE: Header Block - 256 words

Words	Variable Type	Contains
1	INTEGER	The value 12345 - used as a test to verify that it is a valid plot file.
2	INTEGER	Unused
3	INTEGER	IECC - earth correction option
4	INTEGER	NBLK - number of blocks in the file
5	INTEGER	Number of points computed from data base
6	INTEGER	number of points requested from outside the data base

7-10	INTEGER	Data base file name
11-14	INTEGER	Profile file name
15	INTEGER	NBLK - same as word 4
16-256		Unused
Block 1		
1	INTEGER	Plot type - 6 = contour 7 = line of sight 8 = RTM 9 = 3-D 10 = perspective
2		X-coordinate of plot orgin
3-4		Y-coordinate of plot origin
5-6		X-length of plot
9-10		Y-length of plot
11	INTEGER	Plot Command 1 = PLOT 2 = DASH 3 = NUMBER 4 = SYMBOL 5 = HARD COPY 99= READ NEXT BLOCK -99= END OF FILE

Depending on which plot command, the next N number of words contains the needed parameters (Pen up/Pen down, coordinates, angle, etc). PLOT uses 6 words, DASH uses 6 words, NUMBER uses 12 words, SYMBOL uses 48 words, and COPY uses 2 words. The commands are retrieved in sequence and passed to the Tektronix and Versatec supplied routines. They repeat until end of file, however, no command spans a block boundary.

POINT FILE
"PNTFL"

LUN: 33

READ FROM: FTMAIN,PTRD,FTSWP,HIDDEN

WRITTEN TO: FTMAIN,FTSWP,HIDDEN,PPLOT,TPLOT

PURPOSE: The point file contains the plot information generated by perspective and 3-D programs and which is needed to plot feature data on these types of plots. The process of determining which points are hidden from view depends on the point file.

STRUCTURE: Header record - 256 words

Words	Contains
1-5	FDFIL Common Area
6-9	PLOTVR Common Area
10-12	PTFIL Common Area
13-14	PLTBF Common Area (NPBLK & NWRD only)
15-16	FAC Common Area
17	GRDBLK Common Area
18-24	PARAM Common Area
25-50	Unused
51-87	THRBLK Common Area (if 3-D plot)
or	
51-80	PRSBLK Common Area (if perspective plot)
81-85	PARPRS Common Area (if perspective plot)
86-98	Unused
99-100	X coordinate of plot origin
101-102	Y coordinate of plot origin (PLTSV Common Area)
103-256	Unused

Block 1 - It does not appear that Block 1 is ever written to. Thus the point data begins in Block 2.

Block 2 - end of file

256 words - Individual bits are turned on (set to 1) by subroutine HIDDEN.

VERSATEC SWAP FILE
"V2SFL"

LUN: 42

ꓤEAD FROM: VPL2

WRITTEN TO: V2SWP

PURPOSE: This file is used to transfer Versatec plot parameters from PLTRN.SV TO VPL2.SV.

STRUCTURE:

Words	Variable Type	Contains
1-2	REAL	XSCAL
3-4	REAL	YSCAL
5-6	REAL	SCAL
7	INTEGER	ITYPE - Ty0e of plot (See description of plot file 'PLOTF')
8		Unused
9-10	REAL	SCMIN
11-12	REAL	VSTRIP
13-14	REAL	VSTRPO
15-16	REAL	VXMIN
17-18	REAL	VXMAX
19-20	REAL	VYMIN
21-22	REAL	VYMAX
23	INTEGER	IPERM = 0 if plotting from 'PLOTF'; = 1 if from permanent file.
24-27	INTEGER	IFPERM - Plot file name

Note - The variables in words 1-22 take on different values depending on the type of plot being generated. See subroutine VPRMS for the computation of these variables.

IV. COMMON BLOCKS

ADP

BFCM,BUF,BUFFER

CLCOMP,CONBLK,CONPRM,CONT,CONTHR,CONTSC,CORE

DBDAT,DBFIL,DESCRI

EARTH

FAC,FDFIL

GRDBLK,GRID

HEADER,HIDCM,HIDE

IBND,IFFLG,IMGCOM,INDEX,INTYPE,IO,IPFL,IPLOPT,IUTM,IVFILE

LOCUTM,LOSBLK

MGBLK

PARAM,PARPRS,PBUF,PCNT,PLOTVR,PLTBF,PLTCO,PLTSAV,

PNTS,POINT,PPLTS,PRFCM,PRFIL,PRPT,PRSBLK,PTBLK,PTFIL,PTFT

REC,RECORD,REFVAL,RIDGE,RTMBLK,RTMPAR

SCALE,SCOM,SEG,SWPCM

TEK,TEKOPT,THRBLK,TKTRNX

UNITS

VPRM

XEXN

ZONEZ

IVA. PROGRAMS SHARING COMMON BLOCKS

COMMON AREA

ADP	FTMAIN, PTRD
BFCM	RMAIN,LMAIN,PLTRN,BUFFPK
BUF	FTMAIN,FTSWP
BUFFER	ADDREC,CHANGE
CLCOMP	RMAIN,LMAIN,PLTRN,DASH,PLOTS,PLOT,SYMBOL
CONBLK	CMAIN,CPLOT,CONPLT,INCON
CONPRM	CMAIN,INCON,INLOS,INPRS,INRTM,INTHRD,PMPRT,WPRFRT
CONT	CMAIN,PMAIN,PRSWP,RMAIN,RTSWP,TMAIN,THSWP,PERSP,RTM, THREED,CON,INCON,INRTM,PLTOPT,PLTPRT,RPLOT,RTMPLT,VCON
CONTHR	SCALTH,GRDTHR,CALCTH
CONTSC	CMAIN,CPLOT,CONTBD,CONTFT
CORE	PTMN,ALT,WINDOW
DBDAT	NALT,ALT
DBFIL	CMAIN,CPLOT,PMAIN,RMAIN,LMAIN,TMAIN,PERSP,PROFL,PTMN,RTM, THREED,INCON,INLOS,INPRS,INRTM,INTHRD,JPRFRT,WPRFRT, PSWAP,SETUP,TITLE
DESCRI	FTMAIN,DRFTTH,THRFT,DRFTPR,PRSFT,CONTFT,DRIVFT,RTMFT,UNPLT

EARTH	CMAIN,PMAIN,PRSWP,RMAIN,RTSWP,LMAIN,LOSPRT,TMAIN,THSWP PERSP,RTM,THREED,ERTOPT,ERTPLT,ERTPRT,INLOS,INPRS,INRTM,INTHRD, LOSPLT,PPLOT,RPLOT,PRSPLT,PSWAP,RPLOT,RTMPLT,THRPLT,TPLOT
FAC	CMAIN,CPLOT,PMAIN,PRSWP,RMAIN,RTSWP,TMAIN,THSWP,PERSP, FTMAIN,GRDTHR,GRDPRS,PTCALC,RTM,THREED,CON,FTSWP,PPLOT, RPLOT,TPLOT,VCON
FDFIL	CMAIN,CPLOT,PMAIN,PRSWP,RMAIN,RTSWP,TMAIN,THSWP,PERSP, FTMAIN,DRFTTH,DRFTPR,RTM,THREED,DRIVFT,FTOPEN,FTSWP,INCON,INPRS, INRTM,INTHRD,RPLOT,SETUP
GRDBLK	PMAIN,PRSWP,RMAIN,RTSWP,TMAIN,THSWP,PERSP,FTMAIN,GRDTHR,GRDPRS, RTM,THREED,FTSWP,GRDRTM,INPRS,INRTM,INTHRD,PLTOPT, PLTPRT,RPLOT
GRID	PROFL,PTSDR,PTS,PTSSEC
HEADER	MAINFT,ADDREC,CHANGE,PRINR
HIDCM	CPLOT,CON,PPLOT,RPLOT,TPLOT,VCON
HIDE	PERSP,FTMAIN,THREED,FTSWP,HIDDEN
IBND	CMAIN,CONPLT,INCON,INTHRD,MGBOUN
IFFLG	MAINFT,MGSET
IMGCOM	MAINFT,PRINFT,PRINR,UTM2MG
INDEX	CMAIN,PMAIN,RMAIN,LMAIN,TMAIN,PTMN,NALT,ALT,WINDOW, INCON,INLOS,INPRS,INRTM,INTHRD,SETUP,TITLE
INTYPE	MAINFT,ADDREC

IO	CMAIN,PMAIN,RMAIN,LMAIN,LOSPRT,TMAIN,MAINFT,DOPARM,PLOTPM,WINDOW,PLTRN,AGAIN,ERTOPT,ERTPRT,INCON,INLOS,INPRS,INRTM,INTHRD,MGBOUN,MGCORD,MGSET,PLTOPT,PLTPRT,PLTSV,PRCEED,JPRFRT,WPRFRT,RDGPRT,RPLOT,SETUP,STAT,TITLE,TPLOT,UTM2MG,VPRMS
IPFL	DOPARM,PLOTPM,PLTRN
IPLOPT	INPRS,INTHRD,PLTOPT
IUTM	LMAIN,LOSPRT,INLOS,LOSPLT
IVFILE	PLTRN,V2SWP
LOCUTM	PMAIN,PRSWP,RMAIN,RTSWP,LMAIN,PERSP,RTM,INLOS,INPRS,INRTM,MGCORD,PRSPLT,RTMPLT
LOSBLK	LMAIN,LOSPRT,INLOS,LOSPLT
MGBLK	CMAIN,PMAIN,RMAIN,LMAIN,TMAIN,MAINFT,MGBOUN,MGCORD,MGSET
PARAM	CMAIN,PMAIN,PRSWP,RMAIN,RTSWP,LMAIN,TMAIN,THSWP,PERSP,PROFL,PTMN,PTS,FTMAIN,DRFTTH,DRFTPR,RTM,THREED,FTSWP,INCON,INLOS,INPRS,INRTM,INTHRD,PLBPFT,PLSPFT,PPLOT,JPRFRT,WPRFRT,RPLOT,RTMPLT,THRPLT,TITLE,TPLOT
PARPRS	PERSP,FTMAIN,SCALPR,DRFTPR,PRSFT,PRSBD,CALCPR,GRDPRS,PRSLIN,FTSWP,PPLOT
PBUF	PRFWRT,JPRFRD,PRFRD
PCNT	PTMN,ALT,NALT
PLOTVR	CMAIN,PERSP,FTMAIN,DRFTTH,DRFTPR,PRSBD,CALCPR,GRDPRS,RTM,THREED,DRIVFT,FTOPEN,FTSWP,PLBPFT,PLSPFT,RDHDFT,RTMBD,RTMFT,SCALFT

PLTBF	CMAIN,CPLOT,PERSP,FTMAIN,VPL2,RTM,THREED,FTSWP,LOSPLT,PL2, PLOTB,PPLOT,PSWAP,TPLOT
PLTCO	VPL2,PL2
PLTSAV	CMAIN,CPLOT,CONPLT,PERSP,RTM,THREED,LOSPLT,PPLOT,PRSPLT, RPLOT,RTMPLT,THRPLT,TPLOT,FTSWP,FTMAIN,SCALTH
PNTS	CMAIN,CPLOT,PMAIN,RMAIN,LMAIN,TMAIN,PERSP,PTMN,NALT,ALT, RTM,THREED,JPRFRT,WPRFRT,PSWAP,STAT
POINT	CPLOT,LMAIN,LOSPRT,PTMN,PTS,PTSSEC,LOSPLT,PPLOT,RPLOT, TPLOT
PPLTS	PL2,WPLOTS
PRFCM	CMAIN,PMAIN,RMAIN,LMAIN,TMAIN,JPRFRT,WPRFRT
PRFIL	CMAIN,PMAIN,RMAIN,LMAIN,TMAIN,PERSP,RTM,THREED,PMPRT,JPRFRT, WPRFRT,PSWAP
PRPT	FTMAIN,PRSBD,GRDPRS
PRSBLK	PMAIN,PRSWP,PERSP,FTMAIN,SCALPR,FTSWP,INPRS,PPLOT,WPRSPLT
PTBLK	FTMAIN,GRDTHR,GRDPRS,PTCALC,PTRD
PTFIL	PERSP,FTMAIN,THRFT,THRBD,GRDTHR,PRSBD,GRDPRS,PTCALC,PTRD,THREED,FTSWP, TPLOT
PTFT	PERSP,THREED,PPLOT,TPLOT
REC	FTMAIN,DRFTTH,DRFTPR,DRIVFT,PLBPFT,PLSPFT
RECORD	ADDREC,CHANGE,PRINR

REFVAL	MAINFT,ADDREC
RIDGE	PMAIN,PRSWP,RMAIN,TMAIN,THSWP,PERSP,THREED,PLTOPT,PPLOT, RDGPRT,JTPLOT
RTMBLK	RMAIN,RTSWP,RTM,GRDRTM,INRTM,RPLOT,RTMPLT
RTMPAR	RTM,RPLOT,RTMBD,RTMFT
SCALE	GRDRTM,RTMPLT
SCOM	FTMAIN,DRFTPR,PRSBD,CALCPR
SEG	FTMAIN,DRFTTH,THRBD,DRFTPR,PRSBD,CONTBD,PLBPFT,RTMBD
SWPCM	CMAIN,CPLOT,PMAIN,PRSWP,RMAIN,RTSWP,LMAIN,TMAIN,THSWP, PERSP,PROFL,PTMN,PLTRN,RTM,THREED,LOSPLT,PPLOT, JPRFRT,WPRFRT,PSWAP,RPLOT,TPLOT,VPRMS,V2SWP
TEK	CMAIN,PMAIN,RMAIN,LMAIN,TMAIN
TEKOPT	CMAIN,RMAIN,LMAIN,PLTRN
THRBLK	TMAIN,THSWP,FTMAIN,SCALTH,THRBD,THREED,FTSWP,INTHRD,SCALFT, THRPLT,TPLOT,CALCTH
TKTRNX	ALFMOD,BUFFPK,CARTN,CLIPT,CWSEND,DRAWA,HOME,INITT,IOWAIT,LINEF, LMAIN,LVLCHT,MOVABS,MVEA,PCLIPT,PLTCHR,PLTRN,PNTMOD,PSCAL,RECOVR, RESCAL,REVCOT,RMAIN,RSET,TOUTST,VECMOD,VWINDO,V2ST,WINCOT, XYCNVT

UNITS	ALT,CMAIN,CONPLT,GRDRTM,INCON,INLOS,INPRS,INRTM, INTHRD,LMAIN,LOSPLT,LOSPRT,PERSP,PLTOPT,PLTPRT,PMAIN,PPLOT, JPRFRT,WPRFRT,PROFL,PRSPLT,PRSWP,PTMN,PTS,RMAIN,RPLOT, RTM,RTMPLT,RTSWP,SETUP,THREED,THRPLT,THSWP,TITLE,TMAIN,TPLOT, WINDOW
VPRM	PLTRN,VPRMS,V2SWP
XEXN	CMAIN,PMAIN,RMAIN,LMAIN,TMAIN,INCON,INLOS,INPRS,INRTM, INTHRD,MGSET
ZONEZ	CMAIN,PMAIN,LMAIN,RMAIN,TMAIN,MAINFT

IVB. DESCRIPTION OF COMMON VARIABLES

ADP Common Area

COMMON/ADP/IADP

Where:

IADP Is used by subroutine PTRD to calculate which block of the point file ("PNTFL") to read into the IPT array. (See also the PTBLK common area).

BFCM Common Area

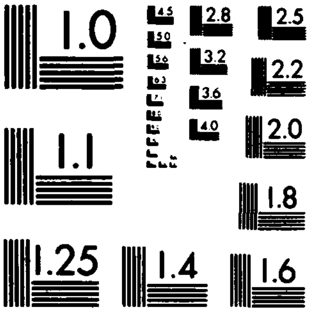
The BFCM common area is a buffer array used by the Tektronix program, subroutine BUFFPK, for generating output to the Tektronix terminal.

COMMON/BFCM/IBFCM(132)

Where:

IBFCM is initialized to zero in DATA statements in BUFFPK,LMAIN,PLTRN, and RMAIN.

AD-A137 977   FEED (FIELD EXPLOITATION OF ELEVATION DATA) SOFTWARE   3/3
DOCUMENTATION(U) GEORGIA INST OF TECH ATLANTA
ENGINEERING EXPERIMENT STATION J G JAY ET AL. AUG 83
UNCLASSIFIED   ETL-0335 DAAK70-82-K-0204   F/G 9/2   NL


1.0
4.5
5.0
5.6
6.3
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

BUF Common Area

The BUF common area stores the values of variables needed to plot feature data on the perspective and 3-dimensional plots.

```
DIMENSION ABUF(128)
COMMON/BUF/IBUF(256)
EQUIVALENCE (IBUF,ABUF)
```

Where:

IBUF The IBUF array is filled with variables from the appropriate common blocks in subroutine FTSWP. The subroutine then writes the array onto block 0 of the Points file.

FTSWP then swaps to "FTMAIN.SV" which reads IBUF from the disk file and puts the values back into the common blocks.

The common blocks involved are FDFIL,PLOTVR,PTFIL,PLTBF, FAC,GRDBLK,PARAM,THRBLK,PRSBLK,PARPRS, and PLTSAV.

BUFFER Common Area

The BUFFER common area is used by the programs which create and modify feature data files.

```
COMMON/BUFFER/IBUF(256)
```

Where:

IBUF Is a 256 word array which is used for temporarily storing information which is to be added to a feature data file.

CLCOMP Common Area

The CLCOMP common area contains information needed by the Tektronix plotting programs.

```
     COMMON/CLCOMP/ICLCMP(31)
or   COMMON/CLCOMP/XSTART,YSTART,XFSET,YFSET,XACUM,YACUM,SKIP,NSKIP,
     IOPT,XLEN,YLEN,XFAC,YFAC,FAC,NHARD,XPOSIT,YPOSIT
```

Where: The variables in the CLCOMP Common Area (except NHARD) are initialized in the Tektronix supplied routine, subroutine PLOTS.

CONBLK Common Area

This common area contains user inputs for the contour plot option. The variables in the CONBLK common area are input or computed in subroutine INCON.

```
COMMON/CONBLK/LDMS(4,2,2),RL(2,2),DLAT,DLON,NPTS,NSCAN,ITIC,NCOPY
```

Where:

LDMS Gives the latitude or longitude of a boundary in degrees, minutes, seconds, and direction as follows:

LDMS(K,1,1)	K=1,4	latitude of southern boundary
LDMS(K,2,1)	K=1,4	latitude of northern boundary
LDMS(K,1,2)	K=1,4	longitude of western boundary
LDMS(K,2,2)	K=1,4	longitude of eastern boundary

LDMS is input by the user in subroutine INCON; it is read from unit number IRD with FORMAT (I3,2I2,A1).

RL Gives the locations of the boundaries as follows:

If the WGS system is being used (INPUT = 1), RL gives the latitude or longitude or a boundary in signed seconds, with

RL(1,1)	=	latitude of the southern boundary
RL(2,1)	=	latitude of the northern boundary
RL(1,2)	=	longitude of the western boundary
RL(2,2)	=	longitude of the eastern boundary

CONBLK Common Area (continued)

If the mil grid system is being used (INPUT = 2), RL gives the boundaries in meters.

RL(1,1)	=	southern boundary
RL(2,1)	=	northern boundary
RL(1,2)	=	western boundary
RL(2,2)	=	eastern boundary

RL is computed in subroutine INCON, using calls to subroutine DMSSEC (if WGS system) or to subroutine MGBOUN (if mil grid system).

DLAT DLAT gives the latitude interval in seconds (if the WGS system is being used) or the northing interval in meters (if the mil grid system is being used).

DLON DLON gives the longitude interval in seconds (WGS system) or the easting interval in meters (mil grid system).

NPTS Is the number of points along a profile.

NSCAN Is the number of profiles.

NPTS and NSCAN are computed in subroutine INCON.

CONBLK Common Area (continued)

ITIC Is the number of terrain points between tic marks; (zero implies no tic marks).

NCOPY Is the desired number of copies of the plot (10).

CONPRM Common Area

The CONPRM common area stores plot parameters corresponding to the plot file being used; these can be printed on the Tektronix terminal by subroutine PLOTPM.

```
DIMENSION PMBUF(128)
COMMON/CONPRM/IPMBUF(256)
EQUIVALENCE (IPMBUF,PMBUF)
```

Where:

IPMBUF This array is filled with variables from the appropriate Common blocks by subroutines INCON,INLOS,INPRS,INRTM, or INTHRD depending on which of the five plot types is being made.

In addition, IPMBUF(1) is assigned values as follows, depending on plot type:

1	contour plot
2	line of sight plot
3	3-dimensional plot
4	perspective plot
5	radar terrain mask plot

The array is then written onto block 0 of the parameters file. Subroutine PMPRT or WPRFRT adds the profile file name to IPMBUF.

Subroutine PLOTPM reads this information from the disk file and prints out the parameters on the Tektronix Terminal. (See PLOTPM).

CONT Common Area

```
COMMON/CONT/NCL,ICUR(11),IC(11)
```

Where:

NCL Is the number of contour levels. At the present time the programs use a maximum of 11 contour levels.

ICUR Gives the contour levels in meters.

IC Counter to keep track of which contour level is being plotted and when to add the contour level symbol. See subroutines CON and VCON.

Depending on which type of plot is being made NCL and ICUR are computed on the basis of user input, by

Subroutine	Plot Type
INCON	Contour Plot
INRTM	Radar terrain mask
PLTOPT	Perspective or 3-dimensional plot.

INCON and PLTOPT set NCL = 11 and compute 11 values of ICUR based on the user inputs for the variables ICMIN (minimum contour level) and ICDEL (contour interval). INRTM does likewise except in the case of masking option #3 ("Safe area below a given ceiling"), in which case there may be fewer than 11 contours, depending on the choice of ceiling height.

CONTHR Common Area

The CONTHR common area contains information needed to draw a grid on the 3-dimensional plot.

```
COMMON/CONTHR/B(2,2),DB(2)
```

Where:

B Gives the locations of the plot boundaries. If the WGS system is being used (INPUT = 1), B gives the latitude or longitude of the boundary in signed seconds. If the mil grid system is being used (INPUT = 2), B gives the boundaries in meters.

DB If the WGS system is being used DB is a latitude or longitude interval in seconds. If the mil grid system is being used, DB is a northing or easting interval in meters.

B and DB are computed by subroutine SCALTH based upon values in the THRBLK Common Area.

CONTSC Common Area

The CONTSC common area stores parameters specifying plot dimensions and scale factors for the contour plot; these parameters are used by the programs which plot feature data on the contour plot.

```
COMMON/CONTSC/FL(2,2),B(2),XSCAL,YSCAL
```

Where:

FL Gives the locations of the boundaries as follows:

FL(1,1)	=	southern boundary
FL(2,1)	=	northern boundary
FL(1,2)	=	western boundary
FL(2,2)	=	eastern boundary

If the WGS system is being used, FL is a latitude or longitude value in signed seconds, while if the mil grid system is used, FL has units of meters. FL is computed in subroutine CPLOT by moving elements from the array variable RL (See CONBLK Common Area) into the array FL.

B Gives the horizontal and vertical plot dimensions in inches with

B(1)	=	horizontal dimension
B(2)	=	vertical dimension

XSCAL Is the horizontal scale factor for the plot in inches/signed second (if WGS) or in inches/meter (if mil grid).

CONTSC Common Area (continued)

YSCAL Is the vertical scale factor for the plot in the same units as XSCAL.

B,XSCAL, and YSCAL are computed in subroutine CPLOT using the parameters returned from subroutine CONPLT.

CORE Common Area

The CORE Common Area is used to read portions of polynomial data base files into memory to speed up processing by minimizing disk access.

```
DIMENSION CNDEX(7,4)
COMMON/CORE/ICORE,MXCORE,ICNDEX(14,4),IFAST(2000)
EQUIVALENCE (ICNDEX,CNDEX)
```

Where:

ICORE = 0 if area in question is not in core, therefore must read from disk
=1 if WINDOW has placed the area of the data base in core.

MXCORE Is the maximum number of words which can fit into the core. MXCORE is initialized to 2000 in subroutine PTMN.

ICNDEX Core index in which is stored the boundary coordinates of the areas which have been read from the data base into core.

IFAST Is the 2000 word area in the core which receives the data base data.

DBUAT Common Area

The DBDAT common area is used to read in data from the data base file.

```
COMMON/DBDAT/IRAY(1280,2)
```

Where:

IRAY	Array into which ALT and NALT read elevation values from the data base file. From this array they extract the profiles.

DBFIL Common Area

```
COMMON/DBFIL/IDBFL(4),IDB
```

Where:

IDBFL Is the unique 4 character name for the data base file to be used for the plots.

"XXXX.DB"
where the X's are input by the user.

IDBFL is read from unit IRD, with FORMAT(2A2) in subroutine SETUP.

IDB Specifies whether or not the program should use a new data base file for the plot, as follows:

0 = No DB file has been opened yet.
1 = A DB file is open; ask whether user wants to reuse it.

DESCRI Common Area

COMMON/DESCRI/IDES

Where:

IDES Is used to specify the size of a military unit as follows:

IDES Value	Unit Size	Symbol Used
1	squad	filled in square
2	between squad and platoon	two filled in squares
3	platoon	three filled in squares
4	company	one vertical line
5	batallion	two vertical lines
6	group or regiment	three vertical lines
7	brigade	one 'X'
8	division	two X's
9	corps	three X's

IDES should be input when feature data records are added or changed. It has not been fully implemented yet. See description of subroutine UNPLT.

IDES is used by SPSYM and SZPLOT. Its value is passed to these programs in an argument.

EARTH Common Area

This common area contains variables used in the earth correction option.

```
COMMON/EARTH/IECC,COEF
```

Where:

IECC Is used to specify the elevation correction option, with:

0 = earth curvature and atmospheric refractivity correction

1 = earth curvature correction

2 = no correction

IECC is input by the user in subroutine subroutine ERTOPT.

COEF takes on the following values:

0	when option 2 is applied
1.0/12756270.0	when option 1 is used
.75/12756270.0	when option 0 is used

FAC Common Area

```
COMMON/FAC/FAC
```

Where:

FAC — Gives the resolution of the Tektronix screen in pixels per inch.

FAC is initialized to a value of 64.0 (i.e. 1024/16) by subroutines CMAIN,PMAIN,RMAIN and TMAIN.

FDFIL Common Area

```
COMMON/FDFIL/IFEAT,IFDFL(4)
```

Where:

IFEAT Indicates whether feature data is to be drawn on the plot. IFEAT is set to 1 in subroutine SETUP if user chooses to plot feature data.

IFDFL Is the unique 4 character file name for the feature data base file.

"XXXX.FD"
where the X's are input by the user.

IFDFL is read in with FORMAT(2A2) in subroutine SETUP.

GRDBLK Common Area

This common area contains the variables used to specify spacing between grid lines when the grid option is chosen. Grids can be drawn on perspective, radar terrain mask and 3-dimensional plots.

```
COMMON/GRDBLK/IGRID
```

Where:

IGRID Gives the spacing between grid lines. This spacing is in seconds if the WGS system is being used (INPUT = 1) and in meters if the mil grid system is being used (INPUT = 2).

IGRID is input by the user in subroutine PLTOPT.

GRID Common Area

```
COMMON/GRID/IANS
COMMON/GRID/IGRID
```

Where:

IANS (IGRID) Specifies whether or not a gridded data base is being used.

Subroutines PTS and PTSSEC use the variable IGRID to determine which to use of the two subroutines for computing elevations of points along a profile.

IGRID = 1 grid data base; calls NALT
IGRID = 1 not grid data base; calls ALT.

IANS is input by the user in PROFL; it is read from unit number 11 with FORMAT (I1).

HEADER Common Area

The HEADER common area contains file index information for a feature data file. (See also subroutines ADDREC and CHANGE, and the main program MAINFT which are used to create and modify feature data files).

```
COMMON/HEADER/INDEX(19)
```

Where:

INDEX   INDEX(1) to INDEX(4) contains the 4-character name for the feature data file.

INDEX(5) is equal to 'W' if the user specifies WGS units or 'U' if UTM units.

INDEX(6) to INDEX(15) is a 20 character or less file description.

These elements of the INDEX array are input by the user in MAINFT.

INDEX(16) is the number of single point records in the file.

INDEX(17) is the number of boundary records in the file.

INDEX(18) is the number of disk blocks in the file.

INDEX(19) is the number of records in the file.

HIDCM Common Area

The HIDCM common area is used to record whether points along a profile are hidden from view.

```
COMMON/HIDCM/IHID(256,2)
```

Where:

IHID   Array elements hold 1 if point is hidden, hold 0 if point is visible.

Note: The information stored in this array is also stored in the point file, using subroutine HIDDEN. This parallel information is used only for plotting feature data in perspective.

IBND Common Area

This common area contains boundary values in mil grid form.

```
COMMON/IBND/IBND(2,4)
```

Where:

IBND the value of a boundary (northern, southern, eastern, or western) for the plot, in mil grid form.

IBND(I,1) I=1,2	southern boundary (northing value)
IBND(I,2) I=1,2	northern boundary (northing value)
IBND(I,3) I=1,2	western boundary (easting value)
IBND(I,4) I=1,2	eastern boundary (easting value)

IBND is input by the user in subroutine MGBOUN as one 4 digit number for each of the 4 boundaries. It is read with FORMAT (2A2).

IFFLG Common Area

COMMON/IFFLG/IFFLG

This common is contained in MGSET and initialized in MAINFT, but it is never used for any purpose. It appears that this common should be deleted.

IMGCOM Common Area

The IMGCOM common area contains a mil grid value used in printing the contents of a feature data file.

```
COMMON/IMGCOM/IMG(4)
```

Where:

IMG Gives the location of a feature data point as a mil grid value (represented as 8 characters).

IMG is computed by subroutine UTM2MG and is printed on the terminal by subroutines PRINFT and PRINR.

INDEX Common Area

The INDEX common area contains information about the data base file.

```
COMMON/INDEX/MASTER(14,18),IDUM(4)
```

Where:

MASTER This array contains variables which are used to characterize the 18 regions in a data base file. (See the detailed description of the data base file).

IDUM contains the remainder of the header block beyond the cell definitions. IDUM(1) holds the indicator whether the data base is WGS or UTM.

The MASTER array is read from block 0 of the data base file by subroutines SETUP and PTMN.

INTYPE Common Area

```
COMMON/INTYPE/IDATA
```

Where:

IDATA Specifies the method chosen to input feature data as follows:

IDATA = 'D' data input from the map by using the digitizing tablet

IDATA = 'K' data input by keying in

IDATA is specified by the user in MAINFT.

IO Common Area

```
COMMON/IO/IRD,IWRT
```

Where:

IRD Specifies the unit number from which user input is read. i.e. for input from the Tektronix keyboard IRD = 11.

IWRT Specifies the output unit number. For output to the Tektronix terminal, IWRT = 10.

These variables are initialized by several programs in DATA statements.

IPFL Common Area

```
COMMON/IPFL/IFL(4)
```

Where:

IFL Is the unique 4 character name for the parameters file that contains the desired plot parameters.

"XXXX.PM"
Where the X's are input by the user.

IPLOPT Common Area

This common area contains 3 parameters input in programs PLOAD and TLOAD. After changing to other programs these values are stored in other common blocks (See RIDGE Common Area).

```
COMMON/IPLOPT/ICMIN,ICDEL,IRGE
```

Where:

ICMIN Is the minimum contour level in meters.

ICDEL Is the contour spacing in meters.

IRGE Is used to specify the ridge line option as follows:

IRGE = 0 Do not plot all ridge lines.
IRGE = 1 Plot all ridge lines.

IRGE is automatically set equal to 1 if the user selects the 'range lines only' or 'range lines and grid' options. If the user chooses one of the other options, he is asked whether he wants ridge lines plotted. (See subroutine PLTOPT for a description of these plot options).

IUTM Common Area

This common area contains the locations of the initial and terminal points for the line-of-sight plot.

```
COMMON/IUTM/ICOORD(4,2)
```

Where:

ICOORD   Gives the location of an initial or terminal point for the line-of-sight plot, as follows:

ICOORD(1,1)	-	ICOORD(4,1)	initial point
ICOORD(1,2)	-	ICOORD(4,2)	final point

ICOORD for each point is an 8 digit mil grid value of form (EEEENNNN). It is computed in subroutine INLOS.

IVFILE Common Area

The IVFILE common area is used if a Versatec plot is being made from a permanent plot file; it contains the permanent plot file name.

```
COMMON/IVFILE/IPERM,IFPERM(4)
```

Where:

IPERM Specifies whether a plot file is permanent, as follows:

IPLOT VALUE	
1	Permanent plot file
0	Otherwise

IFPERM Is the unique 4-character name for the plot file being used.

"XXXX.PL"
wnere the X's are input by the user.

IFPERM is read in with FORMAT(2A2) in subroutine PLTRN. IPERM and IFPERM are written to block 0 of the disk file "V2SFL" by subroutine V2SWP.

LOCUTM Common Area

This common area is used to store a location as a UTM value.

```
COMMON/LOCUTM/IA(4)
```

Where:

IA — Is either an observer location (perspective and radar terrain mask plots) or an initial or terminal point to be used in the line of sight plot.

IA is entered by the user in subroutine MGCORD as an 8 digit UTM value in mil grid form (EEEENNNN); it is read from unit number IRD with FORMAT(4A2).

LOSBLK Common Area

This common area contains user inputs for the line-of-sight plot. The variables in the LOSBLK common area are mostly input or computed in subroutine INLOS.

```
COMMON/LOSBLK/LDMS(4,2,2),XL(2,2),NPTS,NCOPY,IPRT,DELTA,ANGLE,GEODIS
```

Where:

LDMS Gives the latitude and longitude of the initial and terminal points of the plot in degrees, minutes, seconds, and direction as follows:

LDMS(K,1,1)	K=1,4	initial point latitude
LDMS(K,2,1)	K=1,4	initial point longitude
LDMS(K,1,2)	K=1,4	terminal point latitude
LDMS(K,2,2)	K=1,4	terminal point longitude

XL Gives the location of the end points as follows:

If the WGS system is being used (INPUT = 1), XL gives these locations in signed seconds, with

XL(1,1)	=	latitude of initial point
XL(2,1)	=	longitude of initial point
XL(1,2)	=	latitude of terminal point
XL(2,2)	=	longitude of terminal point

LOSBLK Common Area (continued)

If the mil grid system is being used (INPUT = 2), XL gives these locations as northing and easting values in meters with

XL(1,1)	=	northing of initial point
XL(2,1)	=	easting of initial point
XL(1,2)	=	northing of final point
XL(2,2)	=	easting of final point

XL is computed in subroutine INLOS, using calls to subroutine DMSSEC (if WGS system) or to subroutine MGCORD (if mil grid system).

NPTS Is the number of points along the line-of-sight profile (2 NPTS 750). If a value of 0 is entered, a default value of approximately 100 meter spacing between points will be used.

NCOPY Is the desired number of output copies. (≤10).

IPRT Is the option for plot and/or tabular print of profile, with

0	profile plot only
1	profile plot and table
2	tabular print only

LOSBLK Common Area (continued)

DELTA Is the distance in meters between points on the line from the initial to the terminal point.

ANGLE Is the azimuth angle in degrees between the initial and terminal points.

GEODIS Gives the geodetic distance in meters between initial and terminal points.

NPTS,NCOPY, and IPRT are input by the user in subroutine INLOS. DELTA,ANGLE, and GEODIS are computed by the main program, LMAIN.

MGBLK Common Area

The MGBLK common area contains parameters used for input data if the user has chosen to employ mil grid units.

```
COMMON/MGBLK/PRFE,PRFN,XEMIN,XNMIN
```

Where:

PRFE Is a "prefix value" associated with a UTM easting coordinate in the following manner:

UTM Easting Value			PRFE Value
0	UTME	10,000	0
100,000	UTME	200,000	1
200,000	UTME	300,000	3
"	"		"
"	"		"
900,000	UTME	1,000,000	9

PRFN Is a "prefix value" associated with a UTM northing coordinate, and varies in the same manner as as PRFE.

XEMIN Locates the smallest possible UTM easting (input by the user) within the mil grid as follows:

For every UTM easting interval of 100,000 XEMIN varies from 0 to 9,999.

XNMIN Smallest possible UTM northing, represented in the same manner as XEMIN.

These four variables are computed by subroutine MGSET based on user inputs. (See also the ZONEZ Common Area).

PARAM Common Area

The PARAM common area contains parameters needed in conversions between latitude-longitude and UTM (Universal Transverse Mercator) coordinates.

```
COMMON/PARAM/ISPHER,IH,IZONE,REAST,RNORTH
```

Where:

ISPHER Specifies which spheroid code is to be used in the computation as follows:

ISPHER Value	Spheroid
1	Clarke 1880 - spheroid
2	International
3	Clarke 1866
4	Bessel
5	Everest
6	Walbeck
7	Southeast Asia
8	Krasovskiy

IH Specifies whether a geographic location is in the northern or southern hemisphere, with

IH Value	Hemisphere
1	Northern
2	Southern

IZONE Is the UTM zone

REAST Is an easting coordinate

RNORTH Is a northing coordinate

ISPHER,IH, and IZONE are computed in subroutine TITLE.

PARPRS Common Area

The PARPRS common area contains parameters needed to plot a grid and/or feature data on a perspective plot.

```
COMMON/PARPRS/NR,AZ1,AZN,DAZ,RMAX,DR,XL(2)
```

Where:

NR Precise use not determined. These are variables
AZ1 calculated by PPLOT and are azimuth values related to tne
AZN radials.

DAZ Is the spacing between radials. (radians)

RMAX Is the total radial length. (meters)

DR Is the spacing along the radial. (meters)

XL XL(1) is the latitude of the observer in signed seconds (if WGS system) or the northing value for the observer (if mil grid system).

XL(2) is likewise the observer longitude (if WGS) or easting (if mil grid).

DAZ,RMAX,DR, and XL are computed in subroutine SCALPR based on values in the PRSBLK common area.

Note: The use of the PARPRS common area is quite confusing, because it is used very differently oy various subroutines as follows:

Program:	Form of Common
PERSP	COMMON/PARPRS/IPRS(5)
PPLUT	COMMON/PARPRS/AZ1,AZN,NR
FTSWP	COMMON/PARPRS/JPRS(5)
FTMAIN,SCALPR	COMMON/PARPRS/NR,AZ1, AZN,DAZ,RMAX,DR,XL(2)

Although no specific bug has been identified, tne differing lengths of this common block and the different order of some of its variables is troublesome. The potential for a bug to exist here seems possible and should be investigated.

PBUF Common Area

The PBUF common area contains the elevations of the points along a profile. These elevations are computed by the programs in PROFL.SV and are written to a disk file by subroutine PRFWRT. They are later read from the disk file (profile file) by subroutines PRFRD or JPRFRD and used by the plotting programs.

```
COMMON/PBUF/MBLK,NWRD,MWRD,IPBUF(256)
```

Where:

MBLK Is the number of the next block of data to be written to (or read from) the disk file.

NWRD Is the size of the buffer being written to (or read from) disk. (256 words)

MWRD MWRD is the number of words filled in the buffer.

IPBUF is the 256 word buffer containing the elevation values.

The elevation values are computed by subroutines ALT or NALT and are stored in the POINT Common Area. These values are written to the disk file on Unit number 21 (profile file) by means of the buffer array IPBUF.

PCNT Common Area

The PCNT common area contains information about how many profile points were computed from a polynomial data base.

```
COMMON/PCNT/NCR,NDB
```

Where:

NCR Is the number of profile points which have been computed from core.

NDB Is the number of points from the data base.

NCR and NDB are computed in subroutines ALT and NALT.

PLOTVR Common Area

The PLOTVR common area contains information needed by the feature data plotting programs.

COMMON/PLOTVR/INPUT,IPLOT,INRREC,ICONV

Where:

INPUT Is used to specify whether the user inputs for the plot have been in the WGS or mil grid system, with

1	=	WGS
2	=	UTM

IPLOT Indicates which type of plot is being made, with

IPLOT VALUE	PLOT TYPE
1	contour plot
2	radar terrain mask plot
3	3-dimensional plot
4	perspective plot

INPUT And IPLOT are computed by subroutine CMAIN,PERSP,RTM, or THREED depending on which type of plot is being made.

INRREC Is the number of records in a feature data file. Subroutine FTOPEN obtains INRREC by reading the header (block 0) of the feature data file. (See the subroutines in MAINFT.SV for the creation and modification of feature data files).

PLOTVR Common Area (continued)

ICONV Specifies the type of conversion to be done between WGS and mil grid units while plotting feature data, with

ICONV VALUE	CONVERSION
0	none
1	UTM to WGS
-1	WGS to UTM

ICONV is computed in subroutine FTOPEN based on information obtained by reading block 0 of the feature data file.

PLTBF Common Area

The PLTBF Common Area is used to write the plot commands from the FEED programs into the plot file. Subroutines PL2 and VPL2 in PLTRN.SV subsequently use this common to read the commands back in from disk.

```
COMMON/PLTBF/NBLK,NWRD,IPBUF(256)
```

Where:

NBLK Current disk block number

NWRD Current word position in block

IPBUF Array in which the commands are stored prior to writing to disk and also into which the commands are later read back in from disk.

PLTCO Common Area

The PLTCO common area is used to process plotting commands by the Tektronix and Versatec plotting programs.

```
COMMON/PLTCO/IPLT(6),IDSH(6),INUM(12),ISYM(48),ICOP(2)
COMMON/PLTCO/IVPLT(6),IVDSH(6),IVNUM(12),IVYSM(48)
```

Where: Subroutines PL2 and VPL2 read plot commands from the plot file (that were created by the FEED versions of NUMBER, PLOT, SYMBOL, etc.). Depending on the type of plot command, the subroutines move the values into the corresponding variables before passing them to the Tektronix and Versatec PLOT,DASH,NUMBER,SYMBOL, and COPY.

PLTSAV Common Area

The PLTSAV common area is used by all five plot types to store the coordinates of the plot origin.

```
COMMON/PLTSAV/XO,YO
```

Where:

XO,YO   Gives the X and Y screen coordinates of the plot origin in inches.

PNTS Common Area

The PNTS common area contains statistics on the use of the data base file by the profile computing programs in "PROFL.SV".

```
COMMON/PNTS/IALT,ICOUNT
```

where:

IALT Is the number of points computed from the data base file.

ICOUNT Is the number of points requested from outside the data base.

IALT and ICOUNT are computed by subroutine ALT (for a polynomial data base) or by subroutine NALT (for a gridded data base). These variables are printed on the Tektronix terminal by subroutine STAT.

POINT Common Area

The POINT common area contains the elevations in meters of the points along a terrain profile.

```
COMMON/POINT/IA(256,2)
```

Where:

IA This array contains the elevations in meters of points along a profile. The IA array is filled by subroutine PTSSEC (contour plot and 3-dimensional plot) or by subroutine PTS (line-of-sight, perspective, and radar terrain mask plots.) For each point along a profile, these subroutines call subroutine ALT (polynomial data base) or NALT (gridded data base) to obtain an elevation for the point.

The IA array is written to the profile file by subroutine PRFWRT for later use by the plotting programs.

PPLTS Common Area

The PPLTS common area contains information needed by the Versatec supplied plotting programs.

```
COMMON/PPLTS/MODL(19),NDOT(19),DENS(19),NMODL,IOPEN,IWORD,MODEL,XFACT,YFACT,
I2FLG,OUT,STRIP,STRIPO,SPACE,SCALE,UNITS,XMIN,XMAX,YMIN,YMAX,XSTART,YSTART,DEN,
NIBS,NSTRIP
```

Where: Some of the variables in the PPLTS Common Area are initialized by VPL2 (i.e. STRIP,STRIPO,XMIN,XMAX,YMIN, and YMAX), which obtains them from block 0 of the disk file "V2SFL". (See also subroutine V2SWP which writes this information to the disk, and the VPRM Common Area).

Other variables are used internally by the Versatec plotting routines.

PRFCM Common Area

The PRFCM common area contains information based upon user inputs about the type of plot being made.

```
DIMENSION PRF(16)
COMMON/PRFCM/IPRF(32)
EQUIVALENCE (IPRF,PRF)
```

Where:

IPRF Contains the following information:

IPRF(1) Appears to no longer be used.

IPRF(2) Indicates which type of plot is being made

IPRF(2) Value	Plot Type
1	line-of-sight plot
2	radar terrain mask plot
3	contour plot
5	perspective plot
6	3-dimensional plot

The remaining elements of the IPRF array contain information based on user inputs made in subroutines INCON,INLOS, INPRS,INRTM or INTHRD depending on which type of plot is being made.

However, the IPRF array is actually filled by the main programs CMAIN,LMAIN, etc. The IPRF array is used by subroutines JPRFRT and WPRFRT which swap to the profile computing programs in "PROFL.SV". These subroutines give the user a choice of using an existing profile file or creating a new one. If he chooses to use an existing file, then the file header (block 0) is read and information in the file header is compared with the values in the IPRF array as a check on whether the profile file is suitable for the type of plot being made.

PRFIL Common Area

```
COMMON/PRFIL/IPRFL(4)
```

Where:

IPRFL   Is the unique 4 character name for the profile file.

"XXXX.PF"
wnere the X's are input by the user.

IPRFL is read in with format (2A2) in subroutine JPRFRT and WPRFRT.

PRPT Common Area

The PRPT common area stores variables needed by subroutine PRSLIN to plot a feature line between 2 positions on a perspective plot.

```
COMMON/PRPT/IC(2),NC(2),P(2,2)
```

Where:

IC Is a 2 dimensional array containing the profile numbers for the 2 positions.

NC Is a 2 dimensional array containing the point number along the profile for the 2 positions.

P Is an array containing the coordinates in inches of the 2 points as follows:

P(1,1)	=	X coordinate
P(2,1)	=	Y coordinate
P(1,2)	=	X coordinate
P(2,2)	=	Y coordinate

IC and NC are computed by subroutines CALCPR,GRDPRS and PRSBD. P is computed by GRDPRS and PRSBD.

PRSBLK Common Area

This common area contains the inputs for the perspective plot option. The variables in the PRSBLK common area are input or computed in subroutine INPRS.

```
COMMON/PRSBLK/LDMS(4,2),XL(2),AZ0,DST,DSTM,DAZ,NPTS,NSCAN,HT,VANG,VEX,NCOPY,ICR
```

Where:

LDMS Gives the latitude or longitude of the observer in degrees, minutes, seconds, and direction, as follows:

LDMS(J,1) J=1,4	latitude of observer
LDMS(J,2) J=1,4	longitude of observer

XL Gives the location of the observer as follows:

If the WGS system is being used (INPUT = 1), XL gives the latitude or longitude of the observer in signed seconds, with

XL(1)	=	latitude of observer
XL(2)	=	longitude of observer

If the mil grid system is being used (INPUT = 2), XL gives the observer's position, with

XL(1)	=	the northing coordinate of the observer
XL(2)	=	the easting coordinate of the observer

PRSBLK Common Area (continued)

AZO is the bearing of the perspective view in degrees.

DST is the total radial length in meters.

DSTM is the spacing along a radial in meters.

DAZ is the spacing between radials in degrees.

NPTS is the number of points along a radial.

NSCAN is the number of radials.

HT is the height of the observer in meters.

VANG is one half the field of view in radians.

VEX is the vertical exaggeration factor. This must be greater than or equal to 1.

NCOPY is the desired number of copies. (10)

ICR is used to specify the plot options chosen by the user as follows:

0 grid lines
1 range lines
2 contour levels
3 range lines with grid lines
4 contour levels with grid lines

LDMS,AZO,HT,DST,DSTM, and DAZ are input by the user in INPRS: they are read from unit number IRD.
NPTS,NSCAN,VANG,VEX, and NCOPY are computed from user inputs during the same subroutine.

PTBLK Common Area

The PTBLK Common Area contains variables used in plotting feature data on perspective and three dimensional plots.

```
COMMON/PTBLK/JA,JZ,JNUM,IPT(2,768)
```

Where: The exact contents of these variables is as yet undetermined. The confusion is related to the unresolved uses of variables in Common Areas PTFIL and PARPRS. JA and JZ are used by PTRD to determine how many blocks to read in from the point file. JNUM is initialized by FTMAIN to the value NPTS/768 which is used in the calculation of JA and JZ. IPT is the array which receives the data from the point file. All of these variables are used by PTCALC to calculate the screen position for a profile point position.

Note: This common is contained in subroutines GRDTHR and GRDPRS, but no use can be determined. It appears that the common block could be removed from these two subroutines.

PTFIL Common Area

The PTFIL Common Area contains data needed to plot feature data for the perspective and 3-D programs.

```
COMMON/PTFIL/NSCAN,NPTS,NBLK
```

Where:

NSCAN   Is the number of profiles.

NPTS   Is the number of points along a profile.

NBLK   Is the disk block number in point file.

NOTE: This common appears to be used differently by different programs. The variable name NSCAN occurs in other common blocks, so often PTFIL uses the array NN(3) for these variables to avoid conflict. See also the PARPRS common area in which the same confusions currently exist.

PTFT Common Area

The PTFT common area contains information which specifies whether a grid or feature data is to be drawn on a perspective or 3-dimensional plot.

```
COMMON/PTFT/IPTF,IADP
```

Where:

IPTF Indicates whether a grid or feature data is to be drawn on the plot, as follows:

0 Neither grid nor feature data
1 Either a grid or feature data

Subroutines PPLOT and TPLOT swap to the feature data plotting programs in "FTMAIN.SV" only if IPTF = 1.

IADP Block counter used to extend the point file by a sufficient number of blocks to store hidden point information for plotted data (See subroutine PPLOT).

REC Common Area

The REC common area contains information from the header block (block 0) of a feature data file.

```
COMMON/REC/IBUF(256),IWD
```

Where:

IBUF Is a 256 word array which is used for temporarily storing information from the feature data file header. The IBUF array is filled by subroutine FTOPEN (for the perspective or 3-dimensional plot types) or by RDHDFT (for the contour or radar terrain mask plot types).

RECORD Common Area

The RECORD common area contains information about a record in a feature data file. (See also ADDREC, CHANGE, and MAINFT which are used for creating and modifying feature data files).

```
COMMON/RECORD/IREC(13),POS(40)
```

Where:

IREC This array describes a feature data record as follows:

IREC(1) - the record number

IREC(2) - the record code

IREC(3) - IREC(12) - A 20-character description of the record

IREC(13) - The number of points in a boundary record

The record code and description are input from the terminal in subroutines ADDREC and CHANGE. The record code is a number from 1 to 99.

POS The POS array contains the location of each point in a feature data file either as a latitude-longitude value in signed seconds or as a northing-easting value.

REFVAL Common Area

The REFVAL common area contains variables which specify the location of reference points used when feature data is being input by the digitizer. (Four reference points must be input).

```
COMMON/REFVAL/XVAL(4),YVAL(4),IXPT(4),IYPT(4),S1X,S2X,S1Y,S2Y
```

Where:

XVAL   Is the longitude of a point in signed seconds (WGS system) or its easting value (UTM system).

YVAL   Is the latitude of a point in signed seconds (WGS) or its northing value (UTM).

XVAL and YVAL are obtained from user inputs by program MAINFT.

IXPT   Gives the X coordinate of a point obtained from the digitizer.

IYPT   Gives the Y coordinate of a point obtained from the digitizer.

IXPT and IYPT are read in from the digitizer by MAINFT.

S1X   Scale factors computed from the inputs
S2X   to rotate, translate, and convert
S1Y   subsequent digitizer inputs.
S2Y

RIDGE Common Area

COMMON/RIDGE/IRDGE

Where:

IRDGE	Is used to specify the ridge line option as follows:		
	IRDGE =	0	Do not plot all ridge lines.
	IRDGE =	1	Plot all ridge lines.

IRDGE is automatically set equal to 1 if the user selects the 'range lines only' or 'range lines and grid' options. If the user chooses one of the other options, he is asked whether he wants ridge lines plotted. (See subroutine PLTOPT for a description of these plot options).

RTMBLK Common Area

This common area contains inputs for the radar terrain masking plot option. The variables in the RTMBLK common area are input or computed in subroutine INRTM.

```
COMMON/RTMBLK/IFLAG,ICH,LDMS(4,2),RL(2),R,DR,BI,BF,DB,NP,NR,IH,IFH,NCOPY
```

Where:

IFLAG IFLAG is used to specify the terrain masking option chosen by the user, as follows:

1	safe area contours
2	acquisition contours
3	safe area below given ceiling
4	fields of fire

ICH ICH is used to specify the cross hatching option, with

0	cross hatching applied
1	no cross hatching

LDMS Gives the latitude or longitude of the observer in degrees, minutes, seconds, and direction, as follows:

LDMS(J,1)	J=1,4	latitude of observer
LDMS(J,2)	J=1,4	longitude of observer

LDMS is input by the user in subroutine INRTM; it is read from unit number IRD with FORMAT (I3,2I2,A1).

RTMBLK Common Area (Continued)

RL Gives the location of the observer as follows:

If the WGS system is being used (INPUT = 1), RL gives the latitude or longitude of the observer in signed seconds, with

RL(1) = latitude of observer
RL(2) = longitude of observer

If the mil grid system is being used (INPUT = 2), RL gives the observer's position, with

RL(1) = the northing coordinate of the observer
RL(2) = the easting coordinate of the observer

RL is computed in subroutine INRTM, using calls to subroutine DMSSEC (if WGS system) or to subroutine MGCORD (if mil grid system).

R R is the radius of coverage in meters.

DR DR is the spacing along a radial in meters.

BI BI is the bearing of the first radial in degrees.

RTMBLK Common Area (Continued)

BF BF is the bearing of the last radial in degrees.

DB DB is the spacing between radials in degrees.

NP NP is the number of points along a radial.

NR NR is the number of radials.

IH IH is the observer's height above the terrain in meters.

IFH IFH is the ceiling height in meters. This is used when IFLAG=3 (masking option 3).

NCOPY NCOPY is the desired number of plot copies.

IFLAG,ICH,LDMS,DR,BI,BF,IH, and IFH are input by the user in subroutine INRTM. RL,R,NP,NR, and NCOPY are computed from user inputs during the same subroutine.

RTMPAR Common Area

The RTMPAR common area contains plot parameters needed by the subroutines which plot feature data on the radar terrain mask plot.

```
COMMON/RTMPAR/IARC,SC,RAD,AZ1,AZ2,XX1(2,2)
```

Where:

IARC Indicates whether or not the radar terrain mask plot is a full circle plot, i.e. whether the difference between the final and initial bearing is 360^0. IARC = 0 if the plot is a full circle; otherwise it is equal to 1.

SC Is the plot scale in inches/meter.

RAD Is the radius of coverage scaled to inches.

AZ1 Is the bearing of the first radial in radians.

AZ2 Is the bearing of the last radial in radians.

XX1 Gives the location of the observer.

These variables are computed by subroutines RPLOT and RTMPLT based on values in the RTMBLK Common Area.

SCALE Common Area

The SCALE common area contains variables needed to draw a grid on the radar terrain mask plot.

```
COMMON/SCALE/SC,XMAX,XMIN,YMAX,YMIN
```

Where:

SC Is the plot scale in inches/meter.

XMAX Is the maximum X coordinate for the plot in meters.

XMIN Is the minimum X coordinate for the plot in meters.

YMAX Is the maximum Y coordinate (meters).

YMIN Is the minimum Y coordinate (meters).

These variables are computed by subroutine RTMPLT.

SCOM Common Area

```
COMMON/SCOM/SCX,SCY
```

Where:

SCX SCY	These are x and y scale factors for plotting feature data on perspective plots and are computed by subroutine DRFTPR as follows:

If the mil grid system is being used (INPUT = 2),

SCX=1
SCY=1

If the WGS system is used (INPUT = 1),

SCY = 30.9372
SCX = 30.9929 x COS(YAV) where YAV = latitude of the observer (converted to radians)

SEG Common Area

The SEG common area contains information needed to plot boundary type feature data on all plot types except line-of-sight. This information concerns the locations of the boundary point to be plotted and the boundary point most recently plotted.

```
COMMON/SEG/IOLD,POLD(2),INEW,PNEW(2),IST,IBD
```

Where:

IOLD Specifies whether the boundary point previously plotted is inside or outside the plot boundaries.

POLD Gives the coordinates of the boundary point previously plotted, with

POLD(1) - X coordinate
POLD(2) - Y coordinate

INEW *Specifies whether the boundary point currently being* plotted is inside or outside the plot boundaries.

PNEW Gives the coordinates of the boundary point currently bing plotted.

PNEW(1) - X coordinate
PNEW(2) - Y coordinate

SEG Common Area (continued)

IST A value of IST = 0 is used to indicate that the point being plotted is the first boundary point.

IBD Undetermined - possibly no longer used.

INEW,IOLD,PNEW and POLD are computed in subroutines CONTBD,PRSBD, RTMBD, or THRBD depending on which type of plot is being made.

SWPCM Common Area

The SWPCM common area is a buffer array used to simplify the transfer of information to and from disk.

COMMON/SWPCM/ICOM(256)

Where:

ICOM This 256-word array is used for temporary storage of information being transferred to or from a disk file. For example, subroutine WPRFRT, before swapping to the profile computing programs in "PROFL.SV", fills the ICOM array with information from the Common blocks, and writes it to a disk file. Subroutine PROFL then reads the information from the disk and moves it back into the Common blocks.

TEK Common Area

COMMON/TEK/IOPT

This common is included in the main programs, but is found in no other routines. Therefore, it appears to be obsolete and should be removed from the source code.

TEKOPT Common Area

```
COMMON/TEKOPT/INUSER,XORIG,YORIG,XLEN,YLEN
```

This common is included in the main programs and its values are initialized in a DATA statement. However, the common is found in no other routines. Therefore, it appears to be obsolete and the programs should be tested to determine whether it can be removed from the source code.

THRBLK Common Area

This common area contains the inputs for the three dimensional plot option. The variables in the THRBLK common area are input or computed in subroutine INTHRD.

```
COMMON/THRBLK/LDMS(4,2,2),RL(2,2),D(2),N(2),IV,IANG,IZMIN,VEX,NCOPY,ICR
```

Where:

LDMS — Gives the latitude or longitude of a boundary in degrees, minutes, seconds, and direction, as follows:

LDMS(K,1,1)	K=1,4	latitude of southern boundary
LDMS(K,2,1)	K=1,4	latitude of northern boundary
LDMS(K,1,2)	K=1,4	longitude of western boundary
LDMS(K,2,2)	K=1,4	longitude of eastern bondary

RL — Gives the locations of the boundaries as follows:

If the WGS system is being used (INPUT = 1), RL gives the latitude or longitude of a boundary in signed seconds, with

RL(1,1)	=	latitude of the southern boundary
RL(2,1)	=	latitude of the northern boundary
RL(1,2)	=	longitude of the western boundary
RL(2,2)	=	longitude of the eastern boundary

If the mil grid system is being used (INPUT = 2), RL gives the boundaries in meters

RL(1,1)	=	southern boundary
RL(2,1)	=	northern boundary
RL(1,2)	=	western boundary
RL(2,2)	=	eastern boundary

THRBLK Common Area (Continued)

RL is computed in subroutine INTHRD, using calls to subroutine DMSSEC (if WGS system) or to subroutine MGBOUN (if mil grid system).

D D(1) is the latitude interval in seconds (if the WGS system is being used) or the northing interval in meters (if the mil grid system is being used).

D(2) is the longitude interval in seconds (WGS system) or the easting interval in meters (mil grid system).

N N(1) is the number of profiles.
N(2) is the number of points along a profile.

IV IV takes on the following values:

1 if JV='N'
2 if JV='S'
3 if JV='E'
4 if JV='W'

JV is the view face (boundary closest to the projection plane) and is input by the user.

THRBLK Common Area (Continued)

IANG IANG indicates the relationship between view face and the observer position. It takes on the following values:

-1 If a rotation of a "JV" pointing vector into a "JA" pointing vector would be clockwise.

+1 If the same rotation would be counterclockwise.

0 If the "JV" pointing and "JA" pointing vectors are parallel.

JA is the direction from which the area is viewed, and is input by the user.

IZMIN IZMIN is the reference elevation in meters.

VEX VEX is the vertical exaggeration factor. This must be greater than or equal to 1.

NCOPY NCOPY is the desired number of plot copies. (Currently, this is set equal to 1.)

ICR ICR is used to specify the plot options chosen by the user as follows:

0 grid lines
1 range lines
2 contour levels
3 range lines witn grid lines
4 contour levels with grid lines

TKTRNX Common Area

The TKTRNX common area contains information needed by the Tektronix supplied plotting programs.

COMMON/TKTRNX/ITKT(79)

Where:

ITKT Work area used internally by the Tektronix routines. The common is also located in LMAIN,PLTRN, and RMAIN presumably to preserve certain values between overlays.

UNITS Common Area

```
COMMON/UNITS/IDATA,INPUT,ICONV
```

Where:

IDATA Gives the type of units the data base is stored in. Its value is stored in the data base header record.

1 = WGS
2 = UTM

INPUT Is used to specify whether the input data will be entered in the WGS or mil grid system.

1 = WGS
2 = mil grid

INPUT is specified by the user in subroutine SETUP.

ICONV Is used to specify the type of conversion between WGS and UTM values, with

0 = both WGS or UTM (no conversion).
1 = convert from UTM to WGS.
-1 = convert from WGS to UTM.

VPRM Common Area

The VPRM common area contains information needed by the Versatec plotting programs in "VPL2.SV".

```
COMMON/VPRM/XSCAL,YSCAL,SCAL,ITYPE,SCMIN,STRIP,STRIPO,XMIN,XMAX,YMIN,YMAX
```

Where: The variables in the VPRM Common Area define Versatec plotting scales, window limits, and stripping factors.

These variables are computed by subroutine VPRMS. They are written to block 0 of the disk file "V2SFL" by subroutine V2SWP for later use by the Versatec plotting programs.

XEXN Common Area

The XEXN common area contains the mininum possible values for a northing and easting for the current run. These values are used when data is being input in the mil grid system (INPUT = 2).

```
COMMON/XEXN/XE,XN
```

Where:

Xe Is the smallest possible UTM easting value for the lower left mil grid.

XN Is the smallest possible UTM northing value for the lower left mil grid.

XE and XN are input by the user in subroutine MGSET.

ZONEZ Common Area

The ZONEZ common area contains information which defines the military grid system which will be used for input data if the user has specified that data will be in mil grid units.

```
COMMON/ZONEZ/X(33),Y(33),MGCOL,MGROW
```

Where:

X — Gives the easting value for the left (western) boundary of each column of the grid. X increases from the value entered by the user for the smallest possible UTM easting value in increments of 100,000.

Y — Gives the northing value for the bottom (southern) boundary of each row of the grid. Y increases from the value entered by the user for the smallest possible UTM northing value in increments of 100,000.

MGCOL — Is the number of mil grid columns. MGCOL is an integer between 1 and 33.

MGROW — Is the number of mil grid rows. MGROW is an integer between 1 and 33.

MGCOL and MGROW are input by the user in subroutine MGSET. This subroutine also computes X and Y.

IVC. LIST OF INITIAL VALUES

NAME OF COMMON BLOCK	VARIABLE NAME	INITIALIZED VALUE	PROGRAMS CONTAINING DATA STATEMENT
TEKOPT	INUSER	1	CMAIN,LMAIN,RMAIN,(W)PLTRN
	XORIG	0.0	" " " " "
	YORIG	0.0	" " " " "
	XLEN	16.0	" " " " "
	YLEN	12.0	" " " " "
PNTS	IALT	0	CMAIN,LMAIN,PMAIN,RMAIN,TMAIN,(J)PTMN
	ICOUNT	0	
DBFIL	IDB	0	CMAIN,LMAIN,PMAIN,RMAIN,TMAIN
IO	IRD	11	CMAIN,LAMIN,PMAIN,RMAIN,TMAIN,MAINFT, DOPARM,(W)PLTRN
	IWRT	10	" " " "
BFCM	IBFCM	132*0	LMAIN,RMAIN,(W)PLTRN
IFFLG	IFFLG	1	MAINFT
PCNT	NCR	0	(J)PTMN
	NDB	0	(J)PTMN
CORE	MXCORE	2000	(J)PTMN

PBUF	MBLK	1	(J)PRFWRT,(J)PRFRD,PRFRD
	MWRD	0	" " "
	NWRD	256	" " "
PARAM	ISPHER	1	(J)TITLE
	RNORTH	1.E7	"
	REAST	1.E6	"

IV. DISK ORGANIZATION OF FEED

There are three disk packs available for use by the FEED system. They are labeled as follows:

L) OPERATIONAL

2) AUXILLIARY

3) ARCHIVE/DEVELOPMENT

The OPERATIONAL disk contains the current executable versions of all the FEED programs, the data base files, and all other files necessary for plotting. All operational analysis and scene generation should be done on this disk. All programs operate in the primary partition (i.e. DSØ) on this disk.

The AUXILLIARY disk also contains copies of the executable FEED programs on DSØ. If it is decided to load data bases on this disk in the future, it may be used in the same way as the OPERATIONAL disk.

The AUXILLIARY disk also contains a subdirectory named FEED83BK, which is a backup copy of the FEED83 development directory described below. See Section VI. Backup Procedures for more information.

The ARCHIVE/DEVELOPMENT disk is of primary importance to programmers. On this disk are stored all the tools needed for software development. The primary partition (DSØ) of the disk contains the ARCHIVE of old FEED software. All the duplicate versions of the source code, old load maps, relocatable files and libraries, and old FEED executables can be found here. They can serve as a resource for programmers to refer to in resolving questions about previous software development.

The DEVELOPMENT portion of the disk is found in a subdirectory named FEED83. This directory contains the current working versions of all FEED software. Set up here are all the links to utilities located on the primary partition which are needed for FEED software development. Also,the programmer will find load macros for each of the FEED executable programs. These files define precisely the versions of the source code which are loaded into each executable. The files are named LO---.MC; i.e. LOCLOAD.MC,LOWTHREED.MC, etc.

All future software development should be done in the FEED83 directory. When new versions of executable programs are developed and tested, the programmers must remember to transfer copies of the .SV and .OL files to the OPERATIONAL and AUXILLIARY disks.

VI. PROCEDURE FOR SYSTEM BACKUP

The importance of system backup cannot be overstated. The source code, relocatables, executables, and data should be backed up to insure that, in the event of hardware failures, the software can be rebuilt as quickly as possible.

Procedures for system backup are simple, but it takes discipline on the part of system users to see that the procedures are periodically performed.

The software in directory FEED83 should be backed up to tape whenever software development takes place using the following commands:

```
DIR FEED83
INIT MTØ
DUMP/A/N MTØ:Ø
```

This RDOS DUMP file on tape should then be loaded into the FEED83BK directory on the AUXILLIARY disk. After changing disks and rebooting, the following commands should be entered:

```
DIR FEED83BK
INIT MTØ
LOAD/A/V/R MTØ:Ø
```

The /R command instructs the LOAD process to replace only those files with a later date, i.e. the files in which work has been performed.

Maintaining both tape and disk backup files provides protection against losing one or the other due to hardware failure.

Tape backup files containing only the FEED executables (-.SV and -.OL) and the FEED utilities (FORT.SV,RLDR.SV, etc) should also be maintained.

Data base files, which are normally quite large, should be backed up individually to tape. For example: DUMP/A/V MTØ:Ø BQ3Ø.DB.

If it is desired to back up important generated plot files from the OPERATIONAL disk, the following command could be used:

```
DUMP/A/V MTØ:Ø-.PL-.PM
```

DA

FIL